

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)
Факультет інтелектуальних інформаційних технологій та автоматизації
(повне найменування інституту, назва факультету (відділення))


Кафедра комп'ютерних наук
(повна назва кафедри (предметної, циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

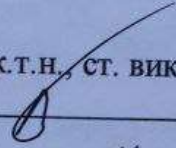
на тему:

**«Інформаційна технологія онлайн помічника для продажу
товарів»**

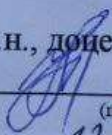
Виконав: студент 2-го курсу, групи 2КН-21м
спеціальності 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки, спеціальності)

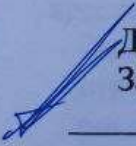

Бортновський О.С.
(прізвище та ініціали)

Керівник: к.т.н., ст. викладач каф. КН


Озеранський В. С.
(прізвище та ініціали)
« 15 » 12 2022 р.

Опонент: к.т.н., доцент кафедри КСУ


Юхимчук М. С.
(прізвище та ініціали)
« 15 » 12 2022 р.


Допущено до захисту
Завідувач кафедри КН

д.т.н., проф. Яровий А.А.
(прізвище та ініціали)
« 16 » 12 2022 р.

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та
автоматизації Кафедра комп'ютерних наук
Рівень вищої освіти II-й (магістерський)
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 122 «Комп'ютерні науки»
Освітньо-професійна програма – «Системи штучного інтелекту»

ЗАТВЕРДЖУЮ
Завідувач кафедри КН
Д.т.н., проф. Яровий А.А.

14.09 2022 року



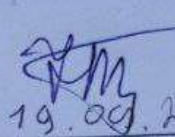
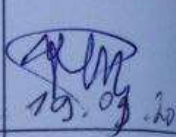
ЗАВДАННЯ

НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Бортновському Олександрю Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Інформаційна технологія онлайн помічника для продажу товарів
керівник роботи к.т.н., старший викладач кафедри КН Озеранський В. С.
затверджені наказом вищого навчального закладу від "14" 09 2022 року № 203
2. Строк подання студентом роботи 18 листопада 2022 року
3. Вихідні дані до роботи:
вхідні дані: індикатори подій користувачів пов'язаних з певним продуктом – не менше 5шт; кількість товарів для аналізу – не менше 100шт; кількість користувачів для одночасної сесії – не менше 500 чоловік.
4. Зміст текстової частини:
Вступ, аналіз сучасного рівня розвитку технології онлайн помічника для продажу товарів, розробка інформаційної технології онлайн помічника для продажу товарів, програмна реалізація інформаційної технології онлайн помічника для продажу товарів, економічна частина, висновки, перелік використаних джерел, додатки.
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень)
Загальний алгоритм функціонування системи рекомендації товарів, загальна структура системи рекомендації товарів, загальна схема роботи системи рекомендації товарів.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконано прийняв
1-3	Озеранський В. С., к.т.н., ст. викл. каф. КН	 14.09.2022	 14.09.2022
4	Буреннікова Н. В., к.с.н., доц каф. ЕПВМ	 19.09.2022	 19.09.2022

7. Дата видачі завдання 14 вересня 2022 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи
1	Аналіз сучасного рівня інформаційної технології онлайн помічника для продажу товарів. Постановка задач дослідження	14.09.2022 р - - 1.10.2022 р.
2	Побудова моделей онлайн помічника для продажу товарів	2.10.2022 р + - 16.10.2022 р
3	Практичне застосування та оцінка ефективності розроблених моделей	17.10.2022 р - - 7.11.2022 р
4	Підготовка економічної частини	8.11.2022 р - - 21.11.2022 р
5	Апробація та/або впровадження результатів дослідження	23.11.2022 р - - 01.12.2022 р
6	Оформлення пояснювальної записки, графічного матеріалу та презентації	02.12.2022 р - - 10.12.2022 р

Студент

Керівник роботи

(підпис)

Бортновський О.С.

(підпис)

Озеранський В.С.

АНОТАЦІЯ

УДК 004.8

Бортновський О. С. Інформаційна технологія онлайн помічника для продажу товарів. Магістерська кваліфікаційна робота зі спеціальності 122 – комп'ютерні науки, освітня програма – комп'ютерні науки. Вінниця: ВНТУ, 2022. 105 с.

На укр. мові. Бібліогр.: 21 назв; рис.: 18; табл. 7.

Дана магістерська кваліфікаційна робота присвячена розробці програмного додатку для продажу товарів. Були розглянуті та проаналізовані існуючі методи продажу товарів онлайн помічниками. Як найбільш перспективний, було обрано реалізацію онлайн помічника на основі нечіткої логіки. Було поставлено задачі необхідні для реалізації даного помічника, такі як збір та аналіз інформації, штучне навчання та рекомендація. На основі існуючих аналогів було визначено та розроблено структуру онлайн помічника, та алгоритм його роботи. Було спроектовано систему рекомендації товарів, написано мовою програмування Python з використанням бібліотек NumPy та Pandas. Точність рекомендацій розробленого додатку на 15% краще аналогу.

Графічна частина складається з 2 плакатів.

У економічному розділі розраховано суму витрат на розробку та виготовлення нового технічного рішення, яка складає 491550 гривень, спрогнозовано орієнтовану величину витрат по кожній з статей витрат, розраховано чистий прибуток, термін окупності витрат для виробника 2,12 років та економічний ефект для споживача при виконанні даної розробки.

Ключові слова: онлайн помічник, штучне навчання, система рекомендацій, продаж товарів, нечітка логіка.

ABSTRACT

Bortnovsky O. S. Information technology of an online assistant for the sale of goods. Master's qualification thesis on specialty 122 - computer science, educational program - computer science. Vinnytsia: VNTU, 2022. 105 p.

In Ukrainian language. Bibliographer: 21 titles; fig.: 18; table 7.

This master's thesis is devoted to the development of a software application for the sale of goods. Existing methods of selling goods by online assistants were reviewed and analyzed. The implementation of an online assistant based on fuzzy logic was chosen as the most promising. Tasks necessary for the implementation of this assistant were set, such as information collection and analysis, artificial learning and recommendation. On the basis of existing analogues, the structure of the online assistant and the algorithm of its operation were determined and developed. A product recommendation system was designed, written in the Python programming language using the NumPy and Pandas libraries. The accuracy of the recommendations of the developed application is 15% better than the analogue.

The graphic part consists of 2 posters.

In the economic section, the amount of costs for the development and production of a new technical solution is calculated, which is 491,550 hryvnias, the estimated amount of costs for each of the cost items is predicted, the net profit is calculated, the payback period for the manufacturer is 2.12 years, and the economic effect for the consumer when implementing this developments.

Keywords: online assistant, artificial learning, recommendation system, sale of goods, fuzzy logic.

ЗМІСТ

ВСТУП	8
1 АНАЛІЗ СУЧАСНОГО РІВНЯ РОЗВИТКУ ТЕХНОЛОГІЇ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ.....	11
1.1 Аналіз та дослідження ринку інформаційних технологій онлайн помічника для продажу товарів.....	11
1.2 Аналіз існуючих технологій онлайн помічника	15
1.3 Постановка задачі онлайн помічника для продажу товарів	22
1.4 Висновок до розділу 1	26
2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ	27
2.1 Обґрунтування вибору інформаційної технології онлайн помічника для продажу товарів.	27
2.2 Постановка задачі	29
2.3 Обґрунтування вибору інструментів для реалізації програмного модуля рекомендації товарів.....	31
2.4 Висновок до розділу 2	33
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ.....	34
3.1 Розробка структури онлайн помічника	34
3.2 Розробка алгоритму роботи онлайн помічника.....	35
3.3 Програмна реалізація інформаційної системи продажу товарів	37
3.4 Тестування та аналіз результатів роботи програмного забезпечення.....	46
3.5 Висновок до розділу 3	49
4 ЕКОНОМІЧНА ЧАСТИНА	50

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки.....	50
4.2 Розрахунок витрат на здійснення науково-дослідної роботи	54
4.3 Оцінювання економічної ефективності науково-технічної розробки.....	63
4.4 Висновок до розділу 4	68
ВИСНОВКИ	69
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	71
Додаток А (обов'язковий) Результат перевірки на плагіат в онлайн-системі UNICHECK	74
Додаток Б (обов'язковий) Лістинг програми	75
Додаток В (обов'язковий) Ілюстративна частина	80
Додаток Г (довідковий) Інструкція користувача.....	83

ВСТУП

Актуальність теми дослідження. В наш час технології не стоять на одному місці. Постійні нові технології, тренди, які здатні достатньо серйозно вплинути на сферу ІТ.

Персоналізація – це ключовий маркетинговий тренд, без якого сьогодні не обходиться ні одна успішна онлайн-компанія. Покупці давно встигли звикнути, що їх улюблені бренди пропонують їм релевантний контент. Тому персоналізовані товарні рекомендації не тільки збільшують продажі, а також зміцнюють відносини з клієнтами.

Онлайн помічник є актуальним і навіть потрібним в наш час. Насамперед це вигідно підприємцям. Завдяки онлайн помічникам для продажу товарів можна оптимізувати витрати та навантаження на робочий персонал. Онлайн помічник має можливість виконувати до 80% задач, які поставлені перед ним.

Онлайн помічник – це розумний бот, який призначений для виконання одноманітної роботи. У порівнянні з людьми, помічник має прискорену реакцію на процеси, і при цьому не втомлюється від монотонних дій. Використання таких онлайн помічників є необхідним для виконання наступних видів операцій: однотипних, швидких та чергування одним з одним. З таким, людина часто не здатна впоратися. Основні причини, для чого потрібні онлайн помічники: для виконання комерційних дій. Великого поширення вони набули сьогодні, отримавши можливість розмовляти з людиною на зрозумілій їй мові. На питання клієнта онлайн помічник дає заздалегідь запрограмовані відповіді. Але для точного вибору потенційних клієнтів веб-персоналізації необхідно збирати дані із всіх точок стосунків з брендом як онлайн, так і в реальності, наприклад, телевізор, рекомендаційна система має враховувати цей факт, щоб не намагатися не продавати вже куплене.

Тема магістерської кваліфікаційної роботи є актуальною, оскільки

завжди існує потреба у онлайн помічниках для продажу товарів, оскільки вони можуть виконувати однотонну роботу за людину у рази швидше та якісніше. Дана технологія буде корисною у бізнесі, для страхових компаній, онлайн банках, інтернет-магазинах або лізингових компаніях, особливо в умовах її незначної вартості придбання, розробки та обслуговування.

Мета та завдання дослідження. Метою дослідження магістерської кваліфікаційної роботи є розширення функціональних можливостей онлайн помічника з продажу товарів за допомогою розробки ефективної рекомендаційної системи.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- Дослідити предметну область онлайн помічника для продажу товарів;
- Дослідити існуючі програми аналоги онлайн помічників;
- Удосконалити інформаційну технологію онлайн помічника для продажу товарів;
- Розробити алгоритм роботи програмного модуля онлайн помічника для продажу товарів;
- Провести програмну реалізацію модуля онлайн помічника;
- Провести тестування розробленого програмного засобу;
- Здійснити економічні розрахунки доцільності розробки нового програмного продукту.

Об'єкт дослідження – це процес продажу товарів за допомогою онлайн помічника.

Предмет дослідження – це програмні засоби продажу товарів за допомогою технології онлайн помічника.

Методи дослідження. За допомогою використання технологій, середі та мови програмування, розробити онлайн помічника для продажу товарів.

Наукова новизна одержаних результатів.

Набула подальшого розвитку інформаційна технологія онлайн помічника з продажу товарів, яка відрізняється від інших використанням удосконаленої моделі рекомендаційної системи, що дозволило розширити функціональні можливості програмного забезпечення з продажу товарів.

Практичне значення одержаних результатів:

- удосконалено алгоритм функціонування он-лайн помічника з продажу товарів;
- розроблено програмне забезпечення он-лайн помічника з продажу товарів.

Апробація результатів роботи. Результати роботи були апробовані на конференції «Молодь в науці» (м. Вінниця, Україна, 2022 р.) [1],

Публікації. За результатами досліджень опубліковано тези доповіді на конференції.

1 АНАЛІЗ СУЧАСНОГО РІВНЯ РОЗВИТКУ ТЕХНОЛОГІЙ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ

1.1 Аналіз та дослідження ринку інформаційних технологій онлайн помічника для продажу товарів

Фільтрування інформації - це процес моніторингу великих обсягів динамічно генерованих даних та виявлення лише того набору цих даних для користувача, який найбільш точно відповідає його потребам.

Онлайн помічник – це автоматизація ручної роботи людини. Це система, бот, штучний інтелект, за якими закріплена задача продажу товару потенційному покупцю. Повідомлення, відправлені з них або на них, оброблюються зовнішньою системою. Крім того, у випадку бота, для користувача спілкування з помічником виглядає як звичайна переписка з реальною людиною.

Онлайн помічник – це розумна програма, яка живе і виконує різні функції.

Функції онлайн помічників:

- Підтримка клієнтів;
- Клієнтський сервіс;
- Маркетинг;
- Робота всередині компанії;
- Рекрутинг;
- Функціональність продвинутих помічників.

З іншого боку, завдання системи рекомендацій - це завдання сфери машинного навчання. У машинному навчанні більшість завдань передбачає навчання функції, яка буде обробляти дані. Для створення такої функції необхідний набір аргументів і значення функції від цих аргументів. В ідеалі, чим більший набір даних, тим точнішою повинна бути функція, але на практиці ми стикаємося з такими проблемами, як шум даних та неправильно

підібраний набір аргументів, що спричиняє проблеми перетренування (overfitting) та недотренування (underfitting). Ці проблеми до кінця не вирішені в рекомендаційних системах. На рисунку. 1.1 можна побачити принцип функціонування рекомендаційної системи.

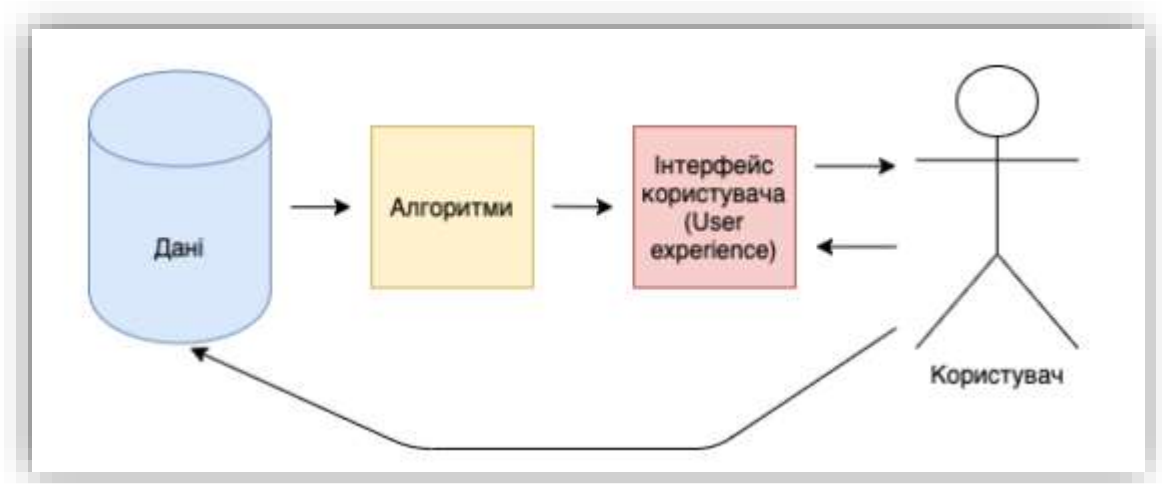


Рисунок 1.1 – Принцип роботи CR

Для виконання своїх повсякденних завдань розробники інтенсивно використовують наявні ресурси, звертаючись до сховищ програмного забезпечення з відкритим кодом. Такі платформи містять багаті джерела даних, наприклад, фрагменти коду, документацію та обговорення користувачів, які можуть бути корисними для підтримки діяльності з розробки.

Протягом останніх десятиліть було запропоновано декілька методів та інструментів, які надають розробникам інноваційні функції, спрямованні на покращення зусиль щодо розробки, економії коштів і продуктивності.

Існує декілька варіантів класифікації онлайн помічників, але проаналізувавши їх, виділимо два види: бізнес-класифікація онлайн помічника у вигляді програмного додатку і класифікація онлайн помічника по технічному типу.

Системи надають розробникам різні артефакти, такі як бібліотеки

сторонніх розробників, документацію про те, як використовувати прийняті API, або відповідні виклики функцій API. Щоб розробити такі рекомендації, були зроблені різні технічні рішення для подолання проблем, пов'язаних з декількома аспектами, включаючи відсутність вихідних даних, обмежену доступність даних, рішення щодо показників ефективності та підходу до оцінки.

Розглянемо кожний тип більше конкретніше:

Розмовні – створені для спілкування, дуже схожі на спілкування з реальною людиною, не маючи конкретної цілі.

Асистенти – виходячи з конкретних цілей, із користувацьких відповідей збирають необхідні дані.

Q&A – принцип роботи, одне запитання – одна відповідь.

Відповідно до досліджень LiveWorld, ще чотири роки тому 58% маркетологів не хотіли вкладувати засоби в онлайн помічників.

Онлайн маркетинг і соціальні мережі набирають популярності останні 10 років: вони були не тільки платформою для спілкування, але й для введення бізнесу. Багато компаній стали популярними завдяки соціальним мережам.

Завдяки новому дослідженню, глобальні продажі штучного інтелекту виростуть до 126 мільярдів доларів до 2025 року порівняно з 10.1 мільярдами в 2018 році. Тепер штучний інтелект бачить своє продовження в чатах. Враховуючи, як бренди застосовують віртуальних асистентів, очікується, що вони стануть основними інвестиціями в обслуговування клієнтів на протязі декількох наступних років.

Системи рекомендацій є важливим компонентом кількох систем онлайн-покупок, що дозволяє власникам бізнесу пропонувати персоналізовані продукти клієнтам. Розробка таких систем завершилася створенням чітко визначених алгоритмів рекомендацій, які, у свою чергу, доводять свою корисність в інших сферах, таких як індустрія розваг або послуги, орієнтовані на працевлаштування.

Рекомендаційні системи в розробці програмного забезпечення були концептуалізовані на порівнянній основі, тобто вони допомагають розробникам орієнтуватися у великих інформаційних просторах і отримувати миттєві рекомендації, які допомагають вирішити конкретне завдання розробки. У цьому сенсі RSSE надає розробникам корисні рекомендації, які можуть складатися з різних елементів, таких як приклади коду, теми, компоненти сторонніх розробників, документація тощо.

Онлайн помічники збільшують продажі. Завдяки їм в короткі терміни можливо отримати цільову аудиторію. При вірному використанні помічника він допоможе у продажі та надасть необхідну інформацію покупцю. Використання чат-бота для підтримки клієнтів допоможе знизити до 40% часу консультації в онлайн-чатах. Окрім цього, до половини звернень до помічника відбувається у неробочий час. На відміну від живої людини, помічник спроможний одночасно давати відповіді декільком користувачам. Більше того, відповідь надається миттєво. Великим плюсом онлайн помічника є кросплатформеність. Готового помічника не важко адаптувати до других платформ. Тому потрібно замовити помічника для Telegram, Facebook, Instagram, WhatsApp або Онлайн-магазину.

Незважаючи на те, що питання розробки та впровадження загальних систем рекомендацій було ретельно розглянуто в сучасних дослідженнях, бракує належних посилань на розробку системи рекомендацій у конкретному контексті, тобто, що задовольняє вимоги різних промислових підприємств. партнери.

Завдяки ретельному дослідженню відповідної роботи ми зрозуміли, що існуючі дослідження стосуються питання проектування та впровадження системи рекомендацій для розробки програмного забезпечення в цілому. Однак, наскільки нам відомо, звіт про досвід, отриманий із реальних проектів розробки, досі відсутній.

Виходячи із вище зазначеного можна зробити висновок, що популярність онлайн помічників набирає обертів, і компанії, які прагнуть до

збільшення ефективності свого бізнесу, все частіше заказують розробникам нових онлайн помічників.

1.2 Аналіз існуючих технологій онлайн помічника

Програмне забезпечення з відкритим вихідним кодом, такі як GitHub та Maven, пропонують багато програмних проектів, які забезпечують стабільні та добре задокументовані продукти. Більшість зазвичай підтримує активні спільноти користувачів і експертів, які, у свою чергу, надають гідну підтримку як для відповідей на запитання користувачів, так і для виправлення повідомлених програмних помилок. Крім того, платформи також є важливим джерелом консультацій для розробників у їхніх щоденних завданнях розробки.

Повторне використання коду є невід'ємною функцією, і розробка нового програмного забезпечення за допомогою наявних компонентів з відкритим кодом дозволяє значно скоротити зусилля на розробку. Вигоди від повторного використання належним чином відібраних проектів з відкритим вихідним кодом численні, включаючи той факт, що система, що впроваджується, спирається на відкритий код. На додаток до вихідного коду, також метадані, доступні з різних пов'язаних джерел, наприклад, каналів зв'язку та систем відстеження помилок, можуть бути корисними для життєвого циклу розробки, якщо їх правильно видобути. Тим не менш, враховуючи безліч джерел даних, розробникам буде важко шукати та підходити до джерел, які відповідають їхнім потребам, не маючи відповідного обладнання.

Такий процес займає багато часу, оскільки проблема полягає не у відсутності, а навпаки, у перевантаженні інформацією, що надходить із різномірних джерел, що швидко змінюються. Зокрема, коли розробники приєднуються до нового проекту, їм зазвичай доводиться освоювати значну кількість джерел інформації. У цьому відношенні розгортання систем, які використовують наявні дані для покращення досвіду розробників, має

першочергове значення.

Для того, щоб почати розробку програмного продукту, необхідно було ознайомитись з ринком онлайн помічників та асистентів для продажу товарів. Кожен помічник виконує певну задачу і працює в своїй предметній області. Оскільки майже всі помічники по своїй структурі є досить простими, наявність інструкції не є необхідністю. Зараз будуть наведені аналоги технологій онлайн помічника для продажу товарів у мережі інтернет:

Чат-боти. Дані боти можуть бути використаними у двох випадках: у месенджерах та безпосередньо на сайті. При цьому данні боти можуть бути як на базі штучного інтелекту, так і готовими відповідями на повідомлення. Даний вид є найпопулярнішим серед онлайн помічників.

Приклади:

1. Чат-бот поштової служби «Justin». Бот допомагає знайти інформацію про відділення, дізнатись про поштові послуги, розрахувати вартість доставки конкретної посилки та відслідкувати її.



Рисунок 1.2 – Чат-бот служби «Justin»

2. Чат-бот лікарні жіночого здоров'я «LabMap». Даний бот допоможе у складанні індивідуального списку необхідних аналізів і розшифруванні їх результатів. Дає можливість зберігати аналізи в одному місці, слідкувати за динамічними показниками і отримувати рекомендації по контролю за здоров'ям. За допомогою бота можна оплатити аналізи у лікарні зі знижкою та здати їх без черги.

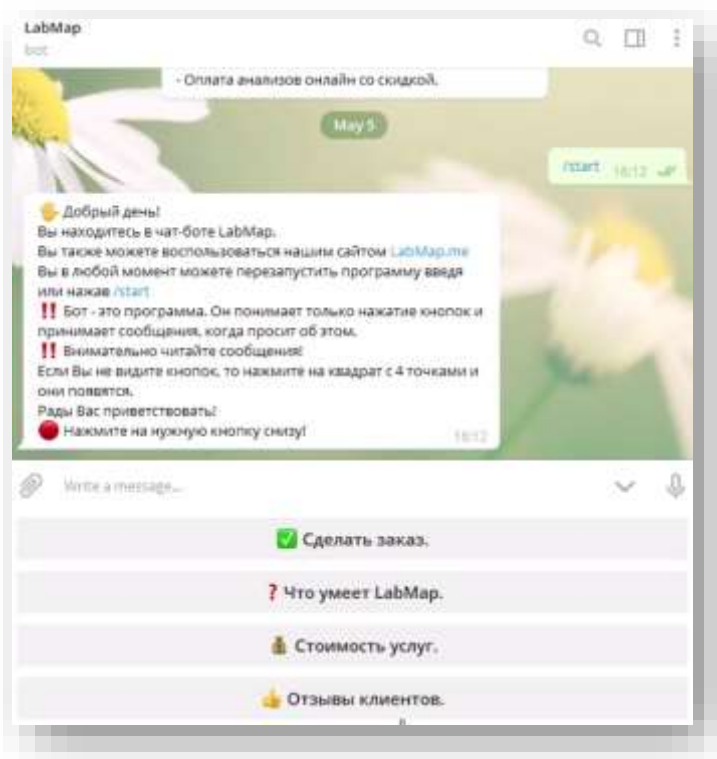


Рисунок 1.3 – Чат-бот лікарні «LabMap»

Голосові помічники. Даний вид онлайн помічників, а точніше асистентів, є досить цікавим. Він реалізований наступним чином: Або напряму на сайті магазину, або окремим додатком, з яким можливо спілкуватись голосом. Така система є складною завдяки тому, що використовується система розпізнавання голосу, система штучного інтелекту.

Приклади:

1. Магазин електроніки «Electronic Store». На даному сайті представлений голосовий асистент, який допомагає у здійсненні покупок. Він може знайти певний розділ, чи товар. Також відкривши картку товару, даний

бот може розповісти його характеристики.

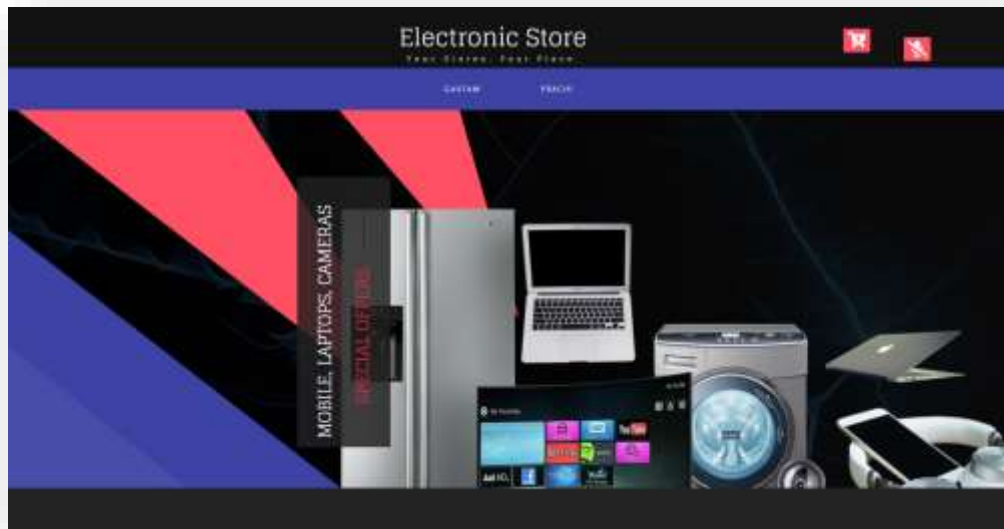


Рисунок 1.4 – Інтернет магазин «Electronic Store»

2. Бот онлайн-асистент «Leon». Це онлайн помічник який спілкується за допомогою голосу та тексту. Він виконує багато функцій, серед яких спілкування, консультація, допомога. Він зможе сказати вам прогноз погоди, новини у вашому місті, чи допомогти у покупках. Даний асистент знайде потрібний вам товар на різних сайтах, чим спростить вам роботу.

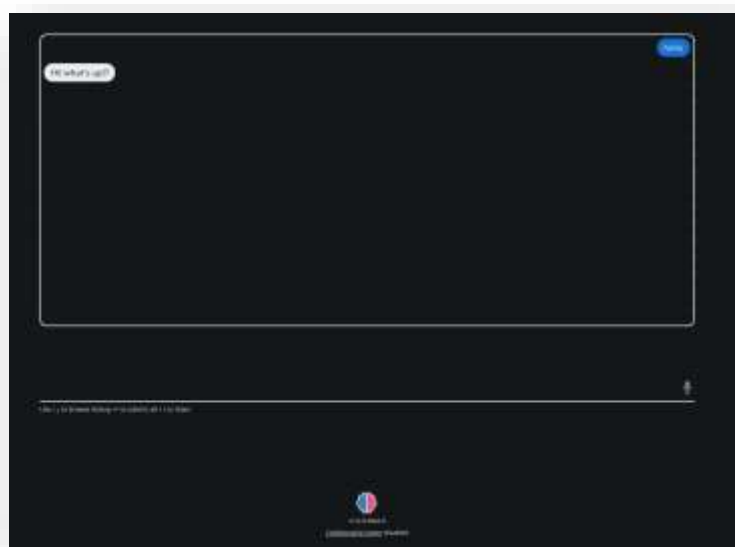


Рисунок 1.5 – Онлайн-асистент «Leon»

Системи рекомендацій. Такі система використовуються на багатьох сайтах онлайн покупок, але вони знаходяться на досить примітивному рівні. Ці системи вмонтовуються у магазин, збирають данні користувачів, їх відгуки, покупки, товари що вони продивляються, і на основі цього рекомендують їм те, що вони з більшою вірогідністю куплять.

1. Система рекомендацій інтернет-магазину «Rozetka».

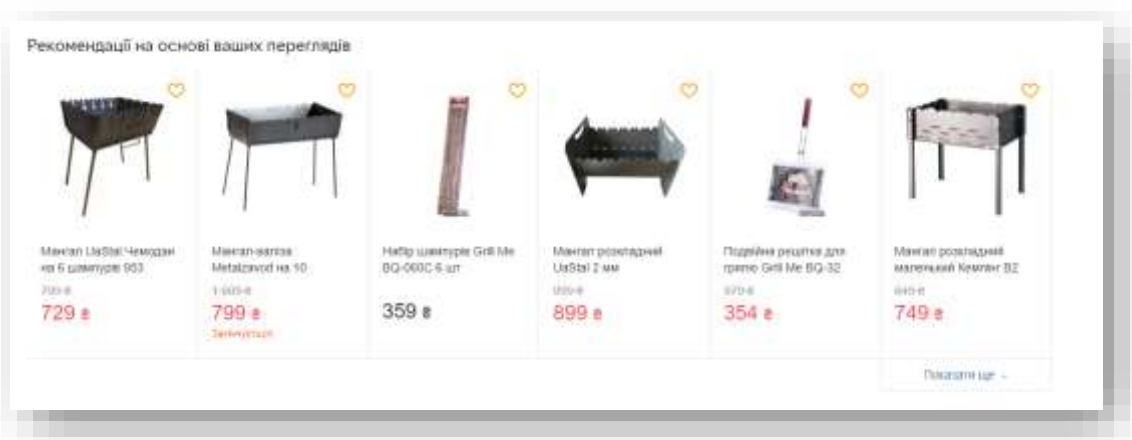


Рисунок 1.6 – Система рекомендацій «Rozetka»

Дана система збирає дані користувачів, включно з моїми. Аналізує їх, дивиться на минулі покупки, ціни і багато іншого, та рекомендує мені якісь товари, особливо акційні. Окрім даної рубрики рекомендацій є і інші.

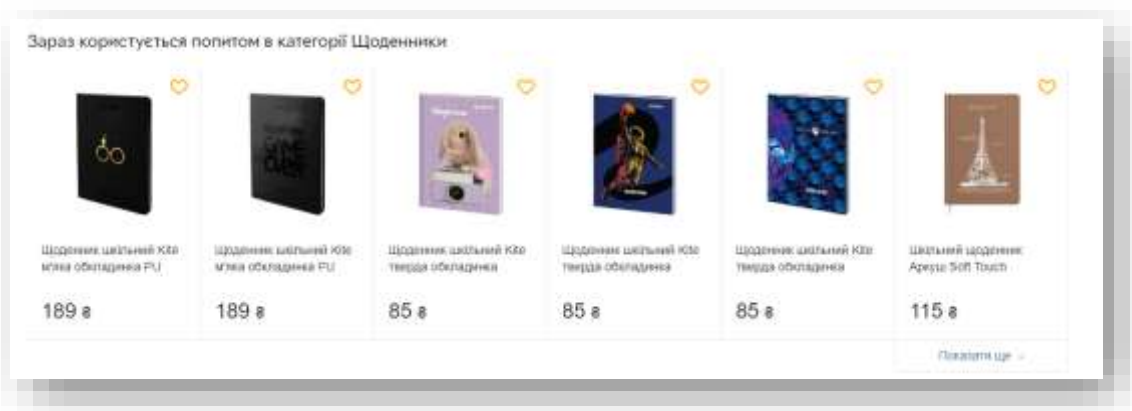


Рисунок 1.7 – Система рекомендацій «Rozetka»

Ми не маємо змогу напряму спілкуватись з даним помічником, але користь від нього така сама як і він чат-ботів та голосових помічників.

Мабуть, одним з найбільш широковідомих прикладів впровадження і використання рекомендаційних систем є компанія Netflix - постачальник відеоконтенту на умовах оренди і у вигляді потокового сервісу.

Компанія починала з того, що розсилала клієнтам по підписці VHS-касети і DVD. Користувач дивився і відправляв диски назад, отримував нові. Для Netflix було важливо підвищити якість рекомендацій. Чим краще Netflix рекомендує користувачам фільми, тим більше фільмів беруть в прокат. Відповідно, зростає і прибуток компанії.

У 2006 році компанія Netflix оголосила конкурс на вдосконалення своєї рекомендаційної системи під назвою Cinematch. Рекомендації формувалися з урахуванням як оцінок користувача, так і оцінок інших глядачів - для цього система підбирала користувачів зі схожими перевагами, чиї оцінки близькі до їх власним. На підставі цього глядачеві автоматично давалася рекомендація: подивитися той чи інший фільм. Власний алгоритм Netflix передбачав оцінок з якістю 0.9514 за метрикою RMSE. Завдання було поліпшити прогноз хоча б на 10% - до 0.8563.

Переможцю обіцяли приз в \$ 1 000 000. Netflix виклав у відкритий доступ зібрані дані: близько 100 мільйонів оцінок за п'ятибальною шкалою з зазначенням ID користувачів, які поставили оцінку. Учасники змагання повинні були якомога точніше передбачати, яку оцінку поставить конкретному фільму той чи інший користувач. Змагання тривало три роки. За перший рік якість поліпшили на 7%, далі все трохи сповільнилося. Проміжні номінації вручалися щороку до тих пір, поки дві команди з невеликою різницею в часі надіслали рішення, кожне з яких проходило поріг в 10%.

Перший приз дістався компанії BellKor's Pragmatic Chaos, групі вчених з AT & T, яким вдалося домогтися поліпшення точності рекомендацій на 10,06%. Компанії продають книги, підписки на музику, фільми, електроніку,

побутові товари, автомобілі, нерухомість та багато інших товарів. Також, з'явилися такі платформи для онлайн навчання як Coursera, Udey, Prometheus. Ці платформи пропонують велике різноманіття курсів на різні тематики, як платних так і безкоштовних. Очевидно, що кількість альтернатив продуктів, які може придбати покупець на веб-сайтах E-Commerce зростає.

Сервіси Amazon, Alibaba пропонують сотні різних товарів, сучасні технології дозволяють обробляти набагато більшу їх кількість. При такому різноманітті користувачу складно підбирати підходящі товари, адже проаналізувати кожен з них фізично неможливо. Такі сервіси мають фільтри пошуку, що дозволяють шукати об'єкти за їхніми атрибутами, але не всі товари мають чітко-виділені атрибути. Наприклад, книги та музика. Добре спроектована рекомендаційна система вирішує цю проблему.

Будь-яка рекомендаційна система допомагає вирішувати певну бізнес-задачу. А результат вимірюється особливими для конкретного бізнес-завдання способами: збільшенням кількості відвідувачів, кількості продаж [2]. Однак якість рекомендаційного алгоритму таким способом виміряти занадто складно, адже бізнес метрики залежать від величезної кількості умов, серед яких рекомендаційний алгоритм сам по собі може виявитися десятою справою. Точність рекомендаційних алгоритмів досить складно оцінити, але існують деякі стандартні метрики, що дозволяють описати точність з математичної точки зору.

Оптимальний підхід, про який піде мова - це спробувати створити метрику безпосередньо під бізнес-завдання. Далі мова піде про конкретний приклад: як це з'роблено для рекомендаційного сервісу кінофільмів imhonet.ru. Хоча цей підхід був перевірений саме на фільмах, але він досить загальний, в чому можна переконатися, замінивши слово фільм на що-небудь ще.

В задачі мова йде саме про список рекомендацій, а не про один елемент. Це тому, що система не в змозі отримати всю інформацію про бажані результати. Навіть пошуковик видає список посилань: він не знає повного

контексту запиту. Якби він знав повний контекст (наприклад, повну інформацію в голові користувача), то, напевно, єдиний результат мав би сенс, а інакше частина роботи все одно залишається на плечах автора запиту. На рисунку. 1.8 зображено можливі варіанти реакції користувача на рекомендації.

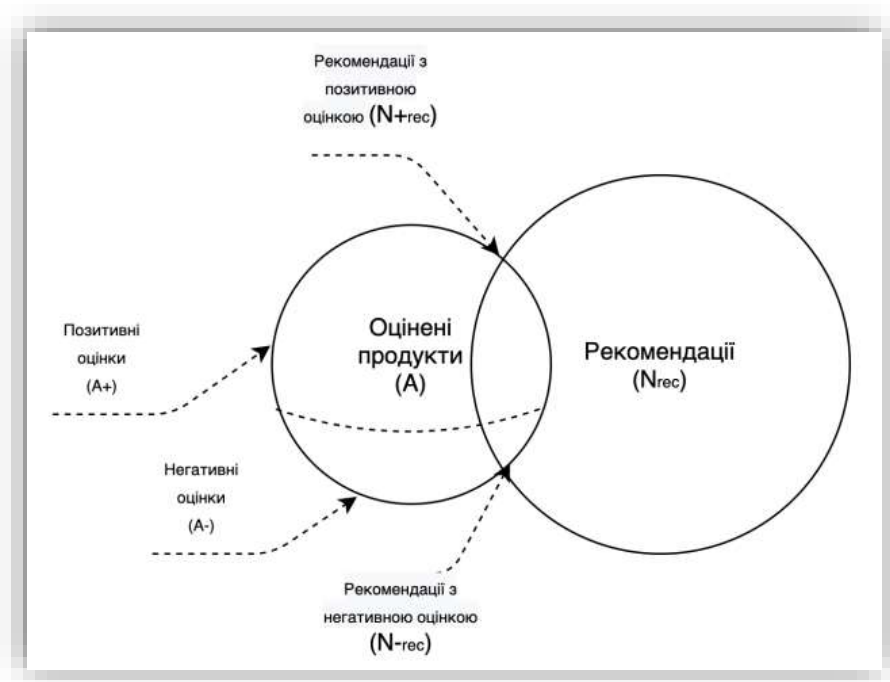


Рисунок 1.8 – Реакція користувача на рекомендації

1.3 Постановка задачі онлайн помічника для продажу товарів

Головна мета – розробити онлайн помічника у вигляді системи рекомендації товарів для продажу товарів онлайн. Даний помічник має можливість виконувати такі функції:

- Аналіз;
- Збільшення кількості проданих товарів;
- Збір інформації;
- Збільшення довіри користувачів;
- Фільтрація інформації;
- Штучне навчання;

- Продаж більшої кількості специфічних предметів;
- Рекомендація товарів для продажу;
- Підвищення задоволеності користувачів;

Дана система може зацікавити не тільки з сторони реалізації, а й з сторони подальшого вдосконалення. Система рекомендацій має необмежені можливості з вдосконалення, наприклад:

Додати більше категорій та розділів;

Додати можливість роботи автономно;

Додати можливість спілкування з системою.

Дана система – помічник, повинена мати зручний інструмент для відображення інформації.

Виходячи з аналізу предметної області, а також аналогів існуючих технологій, можна представити як працює система продажу у сфері послуг. Існуючі аналоги дали зрозуміти, що дана розробка буде актуальною та необхідною у повсякденному використанні.

Користувач взаємодіє з системою через її інтерфейс, яких поділяється на вхідний та вихідний.

- Вхідним інтерфейсом рекомендаційної системи називається сукупність точок взаємодії між користувачем і системою, через які система отримує персоналізовані дані користувача. Введемо поняття рейтингу. Рейтинг – якісна оцінка продукту користувачем, що, зазвичай, виражається числовим значенням. Наприклад, користувач може залишити оцінку відео, яке він переглянув, або залишити коментар під продуктом який він купив. Система отримує персональну інформацію від користувача двома способами: явним (explicit) і неявним (implicit). Вище наведені приклади явного рейтингу, адже користувач ціленаправлено

залишає оцінку, або коментар. Прикладом неявного рейтингу може послугувати ситуація, коли 16 користувач почав дивитися відео і закінчив перегляд за хвилину. Звідси система може зрозуміти, що відео користувачу не сподобалося, адже він завершив перегляд за хвилину.

- Вихідним інтерфейсом рекомендаційної системи називається сукупність точок взаємодії між користувачем і системою, через які користувач отримує рекомендації від системи. Існує поняття передбаченого рейтингу, що визначає з певною ймовірністю передбачену оцінку деякого продукту користувачем. Рекомендацією називають кінцевий набір предметів, який з певною ймовірністю буде до вподоби користувачу. Цей набір може бути відсортований у порядку спаду рейтингу.

Рекомендаційна система і її необхідна користь для покупця та продавця.

Вигоди продавця очевидні, в першу чергу це:

Збільшення лояльності покупця через потрібні, а не настирні рекомендації.

Підвищення доходу за рахунок повторних покупок і продаж схожих товарів.

Розуміння цільової аудиторії, що допомагає більш точно використовувати маркетингові інструменти.

Але і покупець також виграє від використання рекомендацій. Самостійний вибір потребує багато часу: потрібно підібрати необхідні товари, порівнюючи їх між собою і знайти найкращий варіант.

Рекомендаційна же система «відсікає» спочатку непотрібні для покупця речі. В результаті він отримує то, що йому потрібно, з мінімальними затратами у часі.

СР використовують дані різних типів, але їх можливо узагальнити до наступного вигляду: продукти та користувачі.

Продукти є предметами, які рекомендуються. Продукти можуть бути охарактеризованими їхніми властивостями та користю. Наприклад, одяг можливо поділити чоловічий та жіночий, він має колір та розмір. З іншої сторони користувач може бути задоволений, чи розчарований купленим одягом. Це також є характеристикою продукту. РС використовують ці характеристики по різному залежно від алгоритму. Найчастіше, дані про властивості продукту використовуються алгоритмами фільтрування за контентом (CBF), та алгоритмами кластеризації [3]. Властивості продукту називають його ознаками, або опціями. Обов'язковим для продукту являється його ідентифікатор.

Користувач є об'єктом рекомендацій. Для того, щоб ці рекомендації були персоналізованими користувач має деякий профіль. Профіль користувача може містити різноманітну інформацію, як, наприклад, вік та стать. Профілем користувача вважається також список рейтингів, з якими він оцінив деякі продукти. Остання характеристика використовується алгоритмом колаборативного фільтрування (CF). Обов'язковим для користувача є його ідентифікатор. На рисунку. 1.9 зображено відношення між користувачами та продуктами.

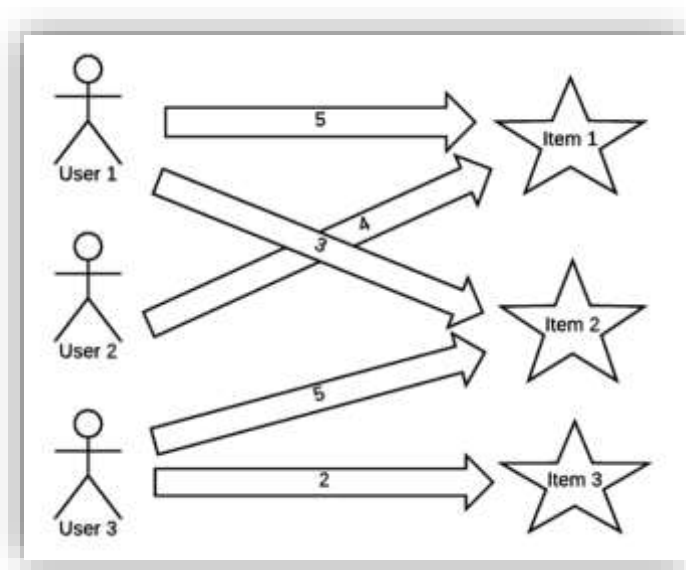


Рисунок 1.9 – Відношення між користувачами та продуктами

1.4 Висновок до розділу 1

У цьому розділі були розглянуті загальні поняття онлайн помічників, ботів, асистентів та систем. Були показані функції їх роботи, наведена класифікація всіх існуючих на даний момент помічників. Також були показані переваги їх використання, на тематиці бізнесу та онлайн маркетингу, де вони застосовуються і в яких компаніях.

Далі була наведена постановка задачі, де визначились, що онлайн помічник повинен робити, які функції мають бути влаштовані, та які межі безпеки потрібно забезпечити для зберігання персональних даних користувачів.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ

2.1 Обґрунтування вибору інформаційної технології онлайн помічника для продажу товарів.

На сьогоднішній день все більше магазинів та підприємств, від маленьких до великих, переходять у онлайн-маркетинг. Роботи у даній сфері є все більше актуальними та потрібними. Тому саме онлайн-маркетинг був вибраний основою даної роботи.

В останні роки діяльність з розробки програмного забезпечення досягла високого ступеня складності, що веде до неоднорідності компонентів, джерел даних і завдань [16]. Запровадження систем рекомендацій у розробці програмного забезпечення (RSSE) спрямоване на підтримку розробників у навігації у великих інформаційних просторах та отримання миттєвих пропозицій, які можуть бути корисними для вирішення конкретного завдання розробки.

Через такий попит, все більше користувачів починають користуватися даними послугами, але не всі вони знають як зробити це без помилок, вірно. Так, більшість магазинів, роблять максимально спрощенні варіанти покупок, так як чим більше людей замовить товар – тим більше грошей отримає компанія. Одним з таких і виступає система для рекомендації та покупки товарів. Вона є досить поширеною, але в той же час досить простою. Це той асистент, з яким неможливо напряму задіяти [4].

У контексті програмного забезпечення з відкритим вихідним кодом розробка нових систем програмного забезпечення шляхом повторного використання існуючих компонентів з відкритим вихідним кодом викликає серйозні проблеми, пов'язані принаймні з такими видами діяльності: (i) пошук компонентів-кандидатів, (ii) оцінка набору отримані компоненти-кандидати, щоб знайти найбільш підходящі, і (iii) адаптація вибраних компонентів

відповідно до певних вимог.

Велика кількість сайтів уже має таку систему, але не всі. Тому потрібно більше приділити увагу ти магазинам, які це не використовують дані технології [18]. Для своєї роботи я обрав книжний магазин. Так як він є досить цікавим у реалізації. Така система може бути схожа на систему підбору фільмів або ігор.

Завдяки сучасним технологіям можливо зробити таку систему більше цікавішою, або навіть вивести її на новий рівень. Розробивши дану систему, її можливо буде впровадити на різні інтернет-магазини електронних книг. Дана система рекомендацій буде виступати у ролі асистента і також онлайн помічника разом. Вона буде збирати дані користувачів: вік, місце проживання, країна, стать і т. д.

Аналізувати їх, за допомогою штучного інтелекту, який у свою чергу за допомогою нечіткої логіки зможе максимально точно підбирати товари (книги) для покупця. Також, якщо удосконалити систему, її можливо зробити універсальною, тобто вона зможе адаптуватись під будь-який сайт. А також додаткову можливість напряду розмовляти з користувачем, для ще більш точнішої рекомендації.

База знань дозволяє розробникам отримати уявлення про необроблені дані, створенні різними інструментами видобутку, а саме:

- Майнери вихідного коду для отримання та зберігання практичних знань із вихідного коду колекції проектів з відкритим кодом;
- NLP-майнери для отримання метрик якості, пов'язаних із каналами зв'язку, і системами відстеження помилок проектів за допомогою обробки природньої мови та методів аналізу тексту;
- Майнер конфігурації для збору й аналізу артефактів і даних конфігурації системи для надання інтегрованого перегляду на рівні DevOps розглянутого проекту з відкритим кодом;
- Міжпроектні майнери для визначення взаємозв'язків між

проектами та додаткові знання, що лежать в основі надання рекомендацій у реальному часі;

Саме тому темою даної роботи було обрано інформаційну технологію онлайн помічника для продажу товарів.

2.2 Поставлення задачі для реалізації онлайн помічника

Можливість надавати рекомендовані елементи за обмежений проміжок часу є важливою, особливо для програм, які потребують миттєвих рекомендацій. Цей показник оцінює тривалість часу, починаючи з моменту, коли користувач надсилає запит, до моменту повернення остаточних рекомендацій.

Залежно від контексту ми повинні вибрати відповідний набір показників для оцінки системи рекомендацій [19]. Наприклад із CrossSim ми можемо використовувати лише показники Succesrate, Confidence, Precision, Ranking I Execution time для оцінки інструменту, оскільки ми поклалися на дослідження користувачів.

Головна мета – розробити систему, асистента, онлайн помічника для продажу товарів. Дана система буде застосована в інтернет-магазині та матиме можливість виконувати такі функції:

- Збір та аналіз інформації;
- Навчання за рахунок штучного інтелекту;
- Рекомендація товарів для продажу.

Дана система може зацікавити не тільки з точки зору реалізації, а й з точки зору подальшого вдосконалення [17]. Даний бот має необмежені можливості з вдосконалення, наприклад:

- Додати більше розділів та категорій;
- Додати можливість розмовляти з штучним інтелектом;
- Додати персональні моделі поведінки.

Дана система повинна мати зручний інструмент для відображення інформації.

Виходячи з аналізу предметної області, а також аналогів, її можна представити як система продажу у сфері послуг. Існуючі аналоги дали зрозуміти, що дана розробка буде актуальною та необхідною у повсякденному використанні.

Основним пунктом продажу є безпека. Це означає, що всі дії, операції, дані, не будуть видимі, не будучи розшифрованими, навіть якщо вони перехоплені [13]. Додатково повинна бути і безпека на усіх інших пристроях. Безпека передачі даних, самий важливий пункт розробки даної системи. Завдяки нечіткої логіки, дана система буде розуміти дані, які використовуються тільки в середині конкретної мережі.

Для вірного визначення задач даного бота потрібно розглянути кожний пункт його роботи. Насамперед даний бот, повинен збирати усі дані сайту та користувачів. Це буде основою для найкращої наступної рекомендацію. А для користувачів, які, наприклад, не зареєструвались, також приділити час. Наприклад система буде поділятися на декілька видів рекомендацій.

Загальна. Даний вид можна представити як головний. Він буде доступний усім користувачам, він буде ґрунтуватись на загальній даних сайту. Які книги у тренді, які книги найкраще купують, книги зі скидками. Це повинно допомогти привернути увагу насамперед нових користувачів, які тільки відвідують даний магазин.

Персональна. Цей вид рекомендацій буде оснований виключно на даних конкретного користувача. Наприклад рекомендація книги, за ціною, країною, та жанром. Даний вид рекомендацій буде направлений на зацікавлення уже конкретного, зареєстрованого користувача [15].

Групова. Даний вид можливо розбити на декілька, таких як акційні пропозиції, пропозиції певного товару, чи певного роду напрямку товарів. Наприклад, якщо якийсь користувач купив предмет А, потім Б, і можливо С, то система побачить це, і коли якийсь користувач також купить предмет А, то

воно порекомендує йому предмет Б, і якщо він зробить дану покупку, то і предмет С також буде рекомендовано. Чим більше даних покупок буде здійснено, тим більш точнішою буде ставати система.

Система буде розвиватись, ставати точнішою, кращою. І з збільшенням користувачів їй буде простіше зрозуміти конкретного користувача, щоб зменшити час на пошук потрібних рекомендацій, та одразу приступити до роботи.

2.3 Обґрунтування вибору інструментів для реалізації програмного модуля рекомендації товарів

Так як мову програмування можна обирати довільну, спочатку потрібно розібратися і дослідити які існують доступні реалізації системи рекомендації товарів і, якщо ні - обрати іншу мову, яка б була максимально зручною для даної реалізації [10].

Так як робота різностороння, вона буде складатися з різних частин. Потрібно визначитись з кожною та обрати найкращу реалізацію.

Для початку поділимо її на безпосередньо реалізацію системи, та її представлення, у вигляді онлайн-магазину електронних книг.

Сама система рекомендацій. Виходячи з даних які потрібні для роботи, для створення системи необхідно використання гібридної нечіткої логіки та генетичних алгоритмів для створення швидкої та точної системи рекомендацій. Для її реалізації найкраще підходить мова програмування Python. З використанням деяких бібліотек, таких як pandas і numpy.

Python – інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією.

Pandas – програмна бібліотека для мови програмування Python для маніпулювання даними та їхнього аналізу.

Numpy – розширення мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою бібліотекою

високорівневих математичних функцій для операцій з цими масивами.

Для представлення даної системи буде розроблено онлайн-магазин електронних книг. Для реалізації даного сайту необхідні такі мови програмування як HTML, CSS, JAVA з використанням BOOTSTRAP.

HTML – стандартизована мова розмітки документів для перегляду веб-сторінок у браузері. Він необхідний для наповнення сайту.

CSS – це спеціальна мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду. Він необхідний для візуалізації.

JAVA – це об'єктно-орієнтована мова програмування. Вона необхідна для анімації сайту.

BOOTSTRAP – безкоштовний набір інструментів, призначений для створення сайтів. Він спрощує розробку динамічних веб-сайтів і необхідний для коректної роботи і відображення на усіх платформах та пристроях.

Система рекомендацій буде розроблена таким чином, щоб вся логіка обрахунків буде зав'язана на використанні бібліотек NumPy та Pandas [11]. Бібліотека NumPy буде використана для роботи з багатомірними масивами, а бібліотека Pandas для обробки та аналізу даних, яка будується поверх бібліотеки NumPy, надає спеціалізовані структури даних і операцій для маніпулювання числових таблиць і непостійними рядами.

Користувач має доступ до баз даних та користувачів, і можливість її редагування. За розділенням контролеру нечіткої логіки стоїть незалежність роботи один від одного для запобігання переплітанням. На рисунку 3.1 можна побачити загальну структурну архітектуру системи:

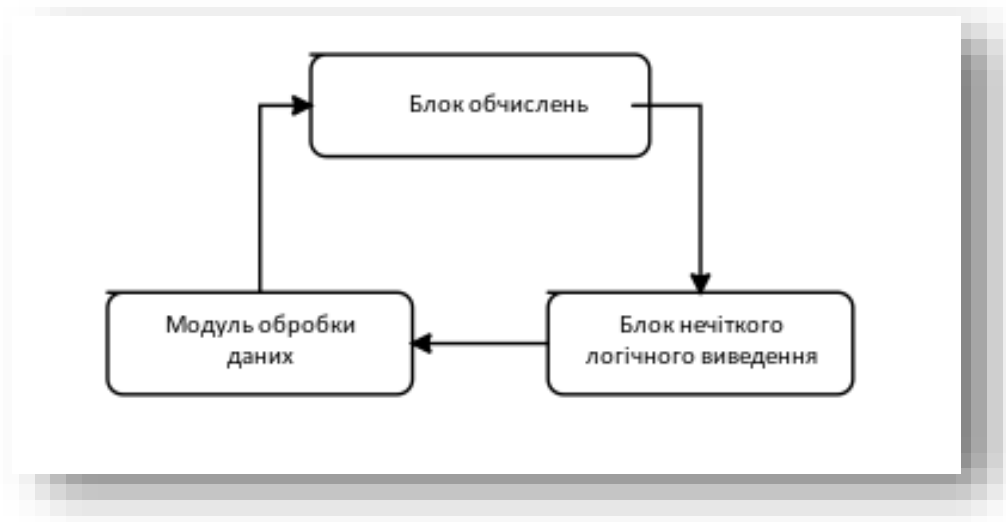


Рисунок 2.1 - Загальна схема функціонування системи

2.4 Висновок до розділу 2

У цьому розділі були розглянуто аналіз методів та інформаційних технологій онлайн помічників, де було розглянуто питання загальної роботи онлайн помічників, ботів, асистентів та їх робота, функції. Також були наведені існуючі онлайн помічники, всіх існуючих видів, їх роботу та реалізацію.

Основним питанням даного розділу було програмна реалізація [9]. Було проаналізовано роботу схожих систем та їх реалізацію, та обгартування вибору мов програмування, а саме мова Python для написання система, з використанням бібліотеки NumPy для роботи з багатомірними масивами, та бібліотеку Pandas для обробки та аналізу даних. Також використання мов HTML, CSS та JAVA для реалізації сайту, онлайн магазину електронних книг, з використанням BOOTSTRAP, для коректного відображення на різних пристроях.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОНЛАЙН ПОМІЧНИКА ДЛЯ ПРОДАЖУ ТОВАРІВ

3.1 Розробка структури онлайн помічника

Система збору вхідних даних. Як було описано у розділі 1 РС має вхідний інтерфейс та вихідний інтерфейс. Ця система має реалізувати вхідний інтерфейс РС користувача, у нашому випадку вхідними даними являються індикатори подій користувача пов'язаних з певним продуктом. Система збору вхідних даних повинна мати інтерфейс для завантаження одиничних індикаторів подій та для великого набору таких даних. Система має швидко поглинати дані та надавати результат операції.

Система обробки даних (тренування моделі). Обробка даних полягає в перетворенні даних індикаторів взаємодії в рекомендації [5]. Існують алгоритми які вирішують цю задачу в режимі реального часу, але на великих об'ємах даних такі алгоритми втрачають свою швидкість або точність. Переважно використовують спосіб тренування моделей офлайн. У проекті також використаний оффлайн підхід тренування моделі [8].

Для розуміння роботи онлайн помічника та її складових необхідно побудувати її структуру. Для розробки системи буде використано мова програмування Python, розроблятиметься в середовищі Visual Studio. Буде задіяно дві бібліотеки NumPy та Pandas.

Система рекомендацій буде розроблена таким чином, щоб вся логіка обрахунків буде зав'язана на використанні бібліотек NumPy та Pandas. Бібліотека NumPy буде використана для роботи з багатомірними масивами, а бібліотека Pandas для обробки та аналізу даних, яка будується поверх бібліотеки NumPy, надає спеціалізовані структури даних і операцій для маніпулювання числових таблиць і непостійними рядами.

Користувач (Адміністратор) матиме доступ до баз даних та

користувачів, і можливість її редагування. За розділенням контролеру нечіткої логіки стоїть незалежність роботи один від одного для запобігання переплітанням. На рисунку 3.1 можна побачити загальну структурну архітектуру системи:

Після схеми функціонування системи рекомендацій, необхідно мати загальну схему роботи усього проекту, який був описаний вище. Зображений на рис. 3.1.

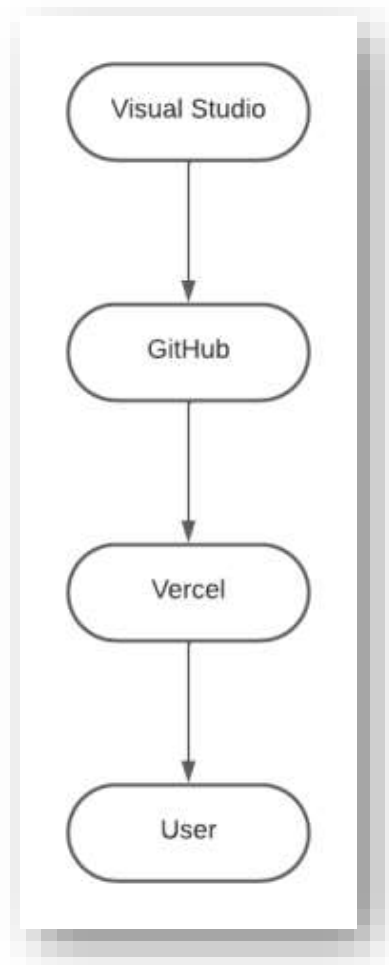


Рисунок 3.1 – Загальна схема роботи програми

3.2 Розробка алгоритму роботи онлайн помічника

Після запуску програмного модуля, система завантажить бази даних користувачів, рейтингу та жанрів [7]. Розпочне тренування користувачів, і на

основі розробленої моделі, з використанням бібліотек почне роботу по рекомендації товарів для конкретного користувача, і не виході ми отримаємо відношення для рекомендації конкретного товару, конкретному користувачеві.

А сам алгоритм функціонування системи представлений на рисунку 3.2.

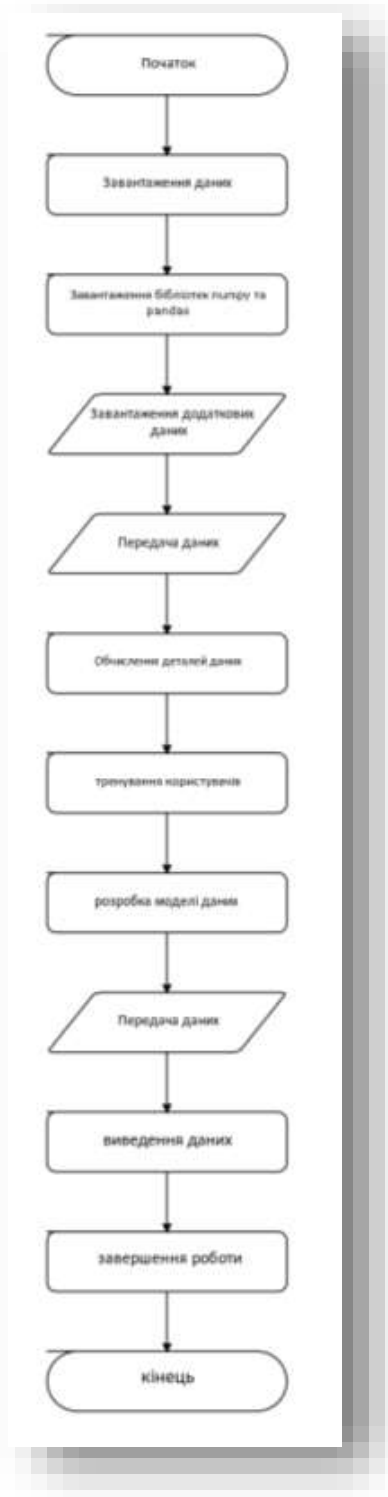


Рисунок 3.2 – Загальний алгоритм функціонування системи

3.3 Програмна реалізація інформаційної системи продажу товарів

Для початку розробки реалізації даної системи потрібно розділи її на частину. Перша частина – це розробка системи рекомендації, друга – це створення сайту-візитки, і третя – це впровадження даної системи, її хостинг та тестування.

Розпочнемо з першої. Система буде розроблятися у середовищі Visual Studio. Установити дану середу можливо на офіційному сайту. Після установки запустимо її, і головне вікно запуску побачимо на рис. 3.3.

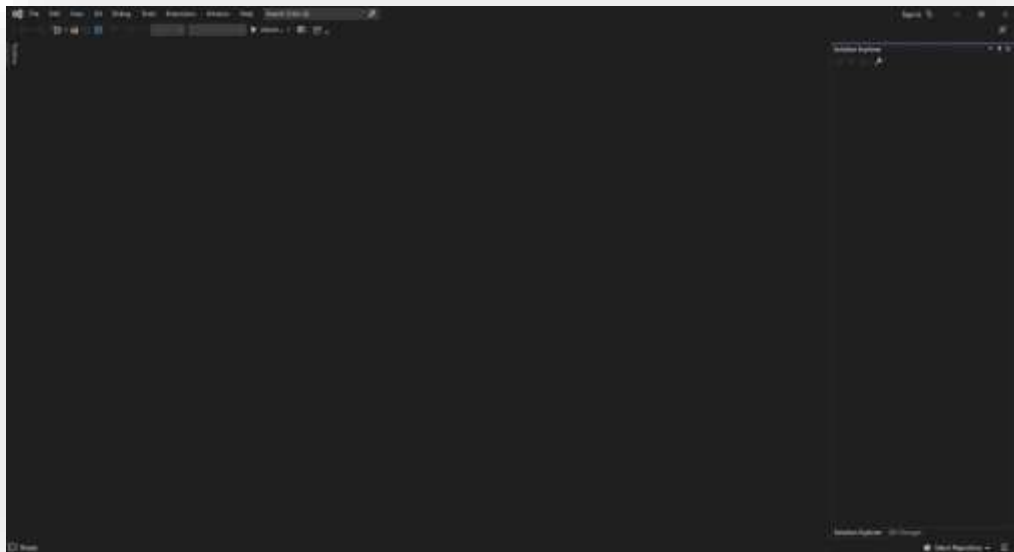


Рисунок 3.3 – Головне вікно запуску Visual Studio

Далі потрібно установити Python IDLE, для роботи та тестування системи, написаної на мові програмування Python. Рис. 3.4.

Для роботи системи потрібно завантажити дві бібліотеки: NumPy та Pandas. Їх установка проводиться за допомогою командної строки. Комбінацією «Windows + R» відкривається вікно виконання. Необхідно написати команду «cmd.exe», тоді нам відкриється командна стрічка від імені адміністратора.

Для установки бібліотеки NumPy необхідно написати наступну команду «python -m pip install numpy», рис. 3.5. Для установки бібліотеки Pandas необхідно написати наступну команду «python -m pip install pandas», рис. 3.6.

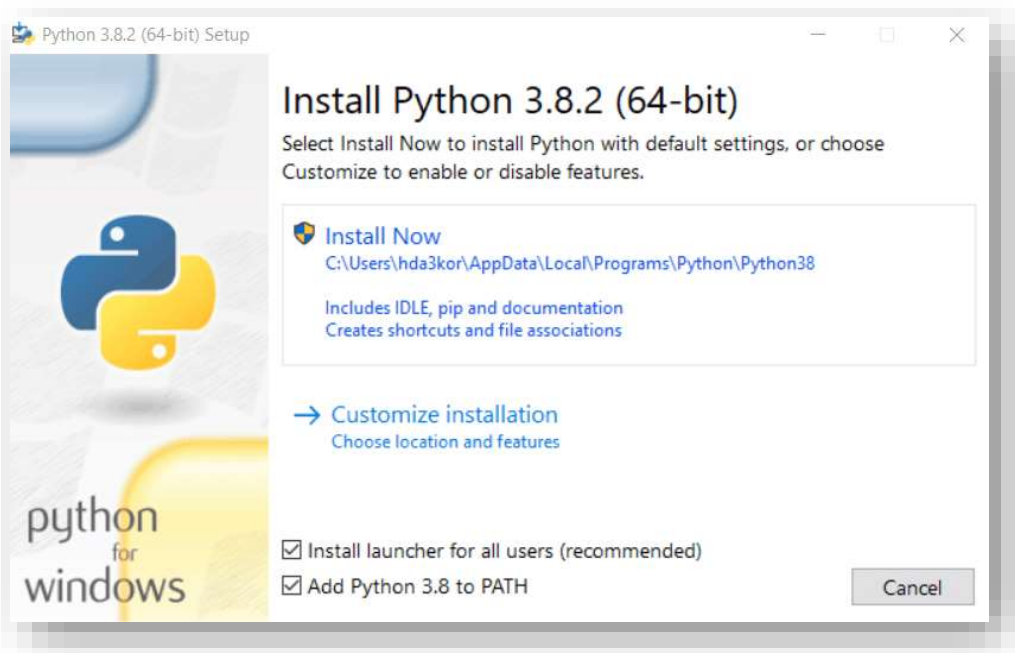


Рисунок 3.4 – Вікно установки Python IDLE

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19041.2006]
(c) Корпорація Майкрософт. Усі права захищені.

C:\Users\bortn>python -m install numpy
C:\Users\bortn\AppData\Local\Programs\Python\Python36-32\python.exe: No module named install

C:\Users\bortn>python -m pip install numpy
Requirement already satisfied: numpy in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (1.19.5)

C:\Users\bortn>
```

Рисунок 3.5 – Вікно установки бібліотеки NumPy

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19043.2086]
(c) Корпорація Майкрософт. Усі права захищені.

C:\Users\bortn>python -m pip install pandas
Requirement already satisfied: pandas in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (1.1-5)
Requirement already satisfied: pytz>=2017.2 in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (from pandas) (2022.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: numpy>=1.15.4 in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (from pandas) (1.19.5)
Requirement already satisfied: six>=1.5 in c:\users\bortn\appdata\local\programs\python\python36-32\lib\site-packages (from python-dateutil>=2.7.3->pandas) (1.16.0)

C:\Users\bortn>

```

Рисунок 3.6 – Вікно установки бібліотеки Pandas

Запускаємо Visual Studio. Відкриваємо репозиторій, де буде зберігатись наш проект. Для початку написання необхідно підключити наші бібліотеки до проекту, рис. 3.7.

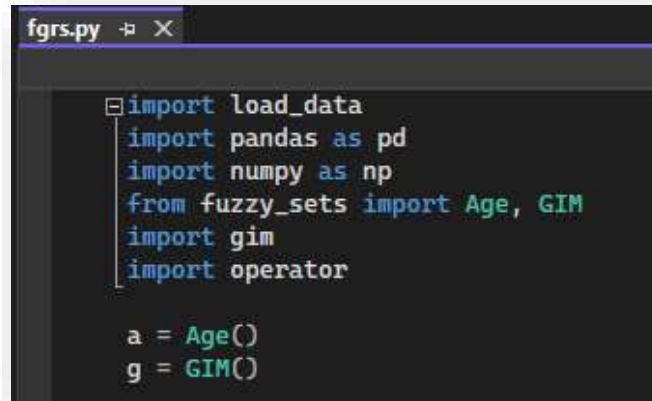
```

fgrs.py
import load_data
import pandas as pd
import numpy as np

```

Рисунок 3.7 – Вікно імпорту бібліотек у проект

Створимо два основні класи. Перший буде збирати дані користувачів, назвемо його «Age». Другий буде виступати показником жанрової цікавості для активних та пасивних користувачів, назвемо його «Gim». Рис. 3.8.



```


fg.rs.py
import load_data
import pandas as pd
import numpy as np
from fuzzy_sets import Age, GIM
import gim
import operator

a = Age()
g = GIM()

```

Рисунок 3.8 – Вікно імпорту бібліотек у проект

Додаємо модуль для збирання user_id, рис 3.9, та невелику частину модуля рекомендацій, рис. 3.10.



```

def fuzzy_distance(ui, uj):
    fuzzy_v = [0] * 21
    for i in range(0, 19):
        ui_gim = [g.very_bad(ui[i]), g.bad(ui[i]), g.average(ui[i]), g.good(ui[i]), g.very_good(ui[i]),
                  g.excellent(ui[i])]
        uj_gim = [g.very_bad(uj[i]), g.bad(uj[i]), g.average(uj[i]), g.good(uj[i]), g.very_good(uj[i]),
                  g.excellent(uj[i])]
        fuzzy_v[i] = dist_func(ui[i], uj[i], ui_gim, uj_gim)
    i = 19
    ui_gim = [a.young(ui[i]), a.middle(ui[i]), a.old(ui[i])]
    uj_gim = [a.young(uj[i]), a.middle(uj[i]), a.old(uj[i])]
    fuzzy_v[i] = dist_func(ui[i], uj[i], ui_gim, uj_gim)
    # додаємо user_id до кожного користувача
    fuzzy_v[20] = uj['user_id']
    # повертаємо fuzzy_v
    return fuzzy_v

```

Рисунок 3.9 – Модуль user_id



```

def recommend(neighbours, testing):
    ms = 0
    for key, item in testing.iterrows():
        ar, m_id = item['rating'], item['book_id']
        n_ratings = []
        for i in neighbours:
            temp = df.loc[df['user_id'] == i].loc[df['book_id'] == m_id]
            for k, it in temp.iterrows():
                n_ratings.append(it['rating'])
            pr = float(sum(n_ratings)) / len(n_ratings) if len(n_ratings) else 0
            ms += abs(pr-ar)
    print(testing.shape[0])
    return ms/testing.shape[0]

```

Рисунок 3.10 – Модуль рекомендацій

Невелика частина коду основного файлу:

```

for i in range(0, 5):
    # active_users and passive_users - pd.Series()
    active_users = top_users.sample(frac=0.10)

    # training_active_users = active_users.sample(frac=0.34)
    # testing_active_users = active_users.drop(training_active_users.index)

    passive_users = top_users.drop(active_users.index)
    # To change this # active users
    tau_data = df.loc[df['user_id'].isin(active_users)][:10]
    index = np.arange(0, tau_data.shape[0])
    i = 0
    for key, value in tau_data.iterrows():
        user_ui_books = df.loc[df['user_id'] == value['user_id']]

        training_user_books = user_ui_books.sample(frac=0.34)
        feature_array = gim.gim_final(training_user_books, value['user_id'])

        # print 'GIM array', feature_array
        feature_array[19], feature_array[20] = value['age'], value['user_id']
        # print feature_array.shape
        model_data_au.loc[i] = feature_array
        i = i + 1
    # print model_data_au
    # print model_data_au
    # Working with passive users
    # change # of passive users

    i = 0
    pu_data = df.loc[df['user_id'].isin(passive_users)][:10]

```



```

for key, value in pu_data.iterrows():
    user_ui_books = df.loc[df['user_id'] == value['user_id']]
    feature_array_p = gim.gim_final(user_ui_books, value['user_id'])
    # print 'GIM array', feature_array
    feature_array_p[19], feature_array_p[20] = value['age'], value['user_id']
    # print feature_array.shape
    model_data_pu.loc[i] = feature_array_p
    i = i + 1
# print model_data_au
# print model_data_pu

# fuzzy_df = pd.DataFrame(columns=m_cols)
fuzzy_vec = []
error = []
for key, value in model_data_au.iterrows():
    i = 0
    for key1, value1 in model_data_pu.iterrows():
        fuzzy_vec.append(fuzzy_distance(value, value1))
        # print value[i], value1[i]
        fuzzy_vec[i] = [(sum(x for x in fuzzy_vec[i][:-1]))**0.5, fuzzy_vec[i][-1]]
        i = i + 1
    # print fuzzy_vec
    neighbours = [n[1] for n in sorted(fuzzy_vec, key=operator.itemgetter(0),
reverse=True)][:30] # taking top 30
    # print neighbours
    testing_user = df.loc[df['user_id'] == value['user_id']].sample(frac=0.66)
    e = recommend(neighbours, testing_user)
    print (e)
    error.append(e)
MAE = sum(error) / len(error)
print (MAE)

```

Тепер друга частина. Після завершення системи, для її роботи потрібно зробити те, для чого вона і створювалась, це сайт [12]. Зробимо сайт – онлайн-магазин електронних книг. Для його розробки необхідно визначитись з середовищем. Найкраще для розробки сайту підходить середовище Sublime Text. Необхідно його встановити, рис. 3.11.



Рисунок 3.11 – Sublime Text

Для наповнення сайту використовуються мова HTML. Частина коду якого представлена на рис. 3.12.



Рисунок 3.12 – Частина коду HTML

Для візуалізації сайту використовується мова CSS. Частина її коду можна побачити на рис. 3.13.



Рисунок 3.13 – Частина коду CSS

Для анімації сайту використовується мова Java. Частина її коду можна побачити на рис. 3.14.

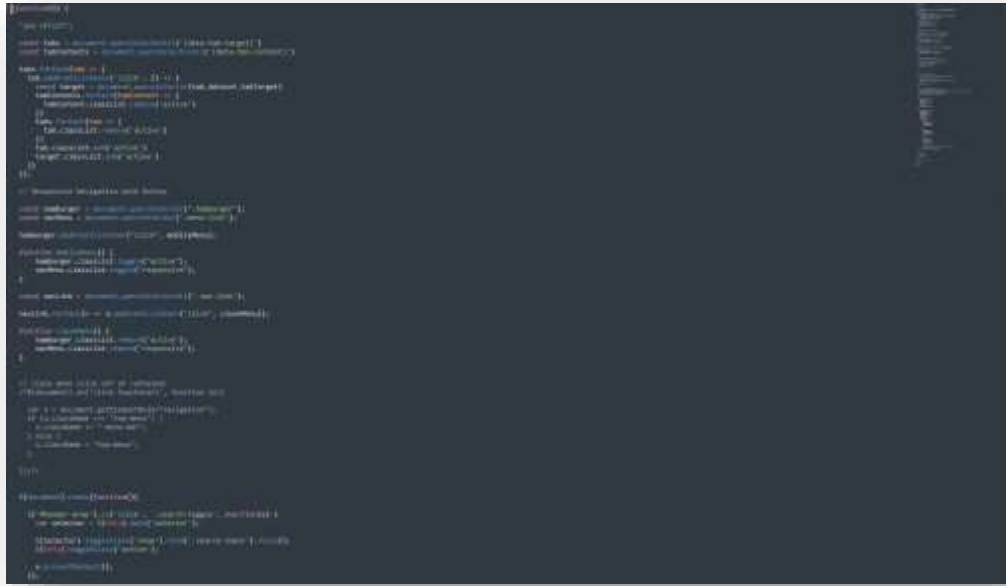


Рисунок 3.14 – Частина коду Java

Остання частина. Фінальною частиною розробки даного проекту є хостинг та запуск. Для цього за допомогою GitHub'а та середовища Visual Studio синхронізуємо код. Створюємо репозиторій, загружаємо наш проект через Visual Studio на GitHub. Хостинг буде проводитись на проекті Vercel. Vercel – безкоштовний хостинг для IT, який дозволяє запускати проекти з GitHub, рисунок 3.15.

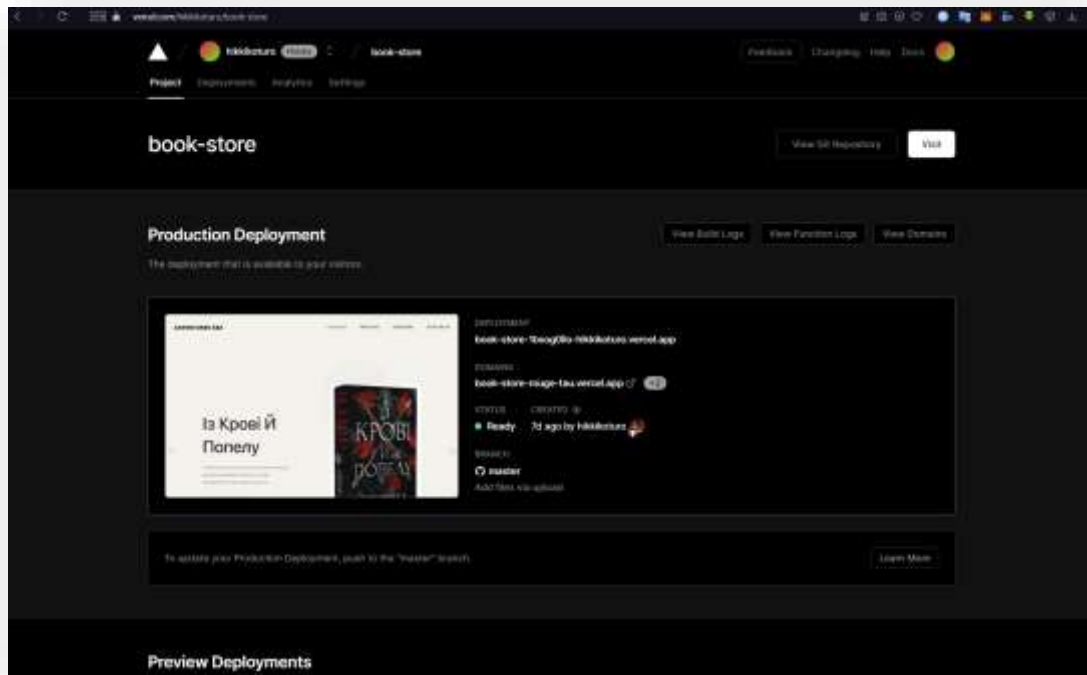


Рисунок 3.15 – Проект-хостинг «Vercel»

3.4 Тестування та аналіз результатів роботи програмного забезпечення

Заключним етапом програмної розробки є тестування. Тут необхідно протестувати дану реалізацію. Зробити це краще на різних пристроях:

ПК/Ноутбук: Для даного тестування візьмемо новий аккаунт (профіль браузера). Та для початку перейдемо на головну сайту. Для перегляду сайту необхідно пароль, рис. 3.16.

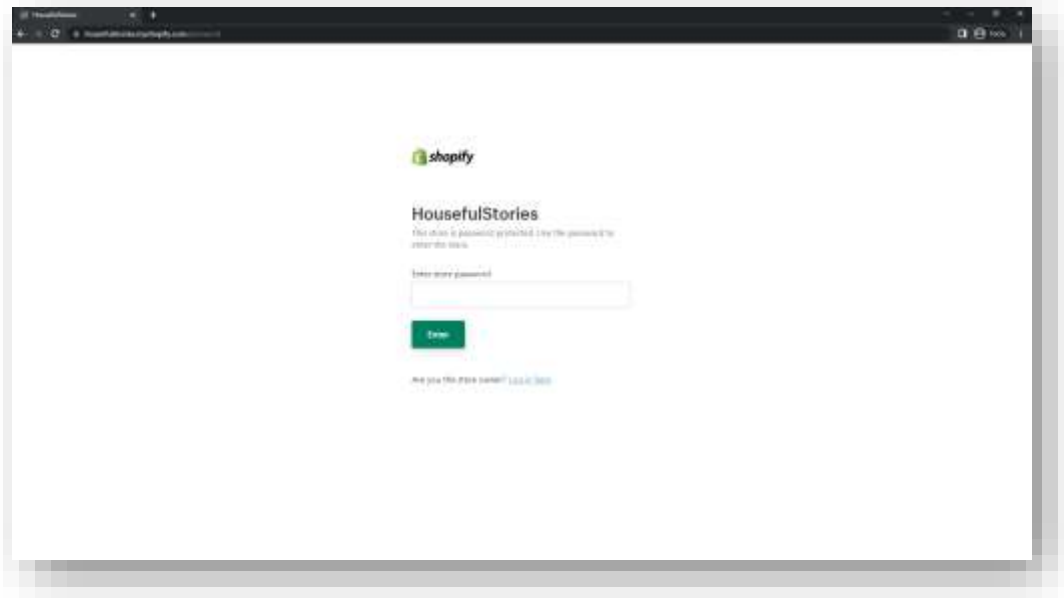


Рисунок 3.16 – Вхід на перегляд сайту

Далі необхідно повністю протестувати сайт, переглянувши всі сторінки та функції. Чи вірно працює відправка повідомлень, також відображення, чи все виглядає як задумувалось, чи відсутні баги [14]. Перевірити наявність помилок та несправностей. І повторити далі дії ще на декількох аккаунтах.

Як результат тестувань, помилок, несправностей та багів виявлено не було. Усі функції які присутні довели свою роботоспособність. Все працює та обробляє швидко. Система з рекомендаціями працює, як без профіля, так і зареєстрованим.

Телефон: Для тестувань використаємо різні пристрої та декілька аккаунтів. Для початку перевіримо чи відображається сайт коректно на мобільних пристроях. Далі необхідно перевірити функції сайту, такі як реєстрація, покупка, система рекомендацій, система рецензій. Також переглянути весь сайт на наявність помилок, багів і т.д. [6].

Та як результат тестувань, різними пристроями та різними аккаунтами, тестування пройшло успішно. Усі функції справні та працюють. Сайт відображається коректно, рис. 3.17.



Рисунок 3.17 – Результати тестування з мобільного пристрою

В результаті розробки було досягнуто поставленої мети за рахунок удосконалення функцій, що є в програмах аналогів, а саме:

- Аналіз масивів даних;
- Розподіл даних за категоріями;
- Пріоритизація завдань;
- Зручний інтерфейс;
- Користувацькі налаштування;
- Збільшення точності рекомендацій;
- Збільшення якості рекомендацій;
- Збільшення кількості користувачів;
- Резервне копіювання;
- Штучне навчання.

Ці функції описані в таблиці 3.1

Таблиця 3.1 – Таблиця порівняння наявності функцій в програмах

	Розподіл по категоріям	Пріоритизація завдань	Резервне копіювання	Штучне навчання	Збільшення якості рекомендацій
Розроблений продукт	+	+	+	+	+
Інтернет магазин «Rozetka»	+	-	+	-	-
Інтернет магазин «Наш формат»	+	+	-	-	-
Netflix	+	-	-	+	+

За результатами таблиці можна побачити, що розроблений програмний засіб має покращений функціонал, принаймні на 2 можливості у порівнянні з програмами-аналогами, який забезпечить вищу продуктивність користувача.

3.5 Висновок до розділу 3

У цьому розділі були розглянуто систему роботи проекту, схему системи рекомендацій, в ході чого, було обґрунтовано і принцип роботи даної системи рекомендацій. Повністю розписано та наведено структуру самого проекту. Показано архітектуру самого проекту, де розповідалось про всі частини та їх значення.

Також важливою частиною даного розділу є програмна реалізація самого проекту, де детально розказано про реалізацію кожного проекту, обґрунтовано вибір мов програмування. Показано детально частини розробки та як воно влаштовані.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності, тобто під час виконання магістерської кваліфікаційної роботи.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п'ятибальної системи оцінювання за 12-ма критеріями, наведеними в таблиці. 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали					
	0	1	2	3	4
1	2	3	4	5	6
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепції підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижча за ціни аналогів	Ціна продукту значно нижча за ціни аналогів

Продовження таблиці 4.1

4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічна та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продуктивності продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Фахівці з питань як з технічної, так і з комерційної реалізації ідеї

Продовження таблиці 4.1

9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більший 5-ти років	Термін реалізації ідеї менший 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менший 3-х років. Термін окупності інвестицій менший 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що потребує часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту потребує незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будьякі регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці, тому була створена таблиця 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки

Критерії	Експерт		
	1	2	3
	Бали:		
Технічна здійсненність концепції	2	2	2
Ринкові переваги (наявність аналогів)	3	3	2
Ринкові переваги (ціна продукту)	2	3	3
Ринкові переваги (технічні властивості)	3	2	2
Ринкові переваги (експлуатаційні витрати)	2	2	3
Ринкові переваги (розмір ринку)	2	1	1
Ринкові переваги (конкуренція)	3	3	3
Практична здійсненність(наявність фахівців)	2	2	1
Практична здійсненність(наявність фінансів)	3	1	2
Практична здійсненність(необхідність нових матеріалів)	3	4	3
Практична здійсненність(термін реалізації)	3	4	3
Практична здійсненність(розробка документів)	2	2	2
Сума балів	30	29	27
Середньоарифметична сума балів	28.6		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використовують рекомендації, наведені в табл. 4.3.

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вищий середнього
21...30	Середній
11...20	Нижчий середнього
0...10	Низький

Результатом рівня та потенціалу роботи стало значення 28.6, що відповідає значенню Середній у таблиці 4.3. Такі показники були досягнуті за рахунок досить середнього рівня та комерційного потенціалу розробки.

4.2 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної, дослідно-конструкторської, конструкторсько-технологічної роботи, створенням дослідного зразка і здійсненням виробничих випробувань, під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуються за такими статтями:

- витрати на оплату праці;
- відрахування на соціальні заходи;
- матеріали;
- паливо та енергія для науково-виробничих цілей;
- витрати на службові відрядження;
- спецустаткування для наукових (експериментальних) робіт;
- програмне забезпечення для наукових (експериментальних) робіт;
- витрати на роботи, які виконують сторонні підприємства, установи і організації;
- інші витрати;
- накладні (загальновиробничі) витрати.

4.2.1 Витрати на оплату праці

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Z_o) розраховують відповідно до посадових окладів працівників, за формулою:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} * t_i}{T_p},$$

де k – кількість посад дослідників, залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – кількість днів роботи конкретного дослідника, дн.;

T_p – середня кількість робочих днів в місяці, $T_p = 21 \dots 23$ дні.

Проведені розрахунки зведено до таблиці 4.4.

Таблиця 4.4 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Кількість днів роботи	Витрати на заробітну плату, грн
Керівник проекту	12075	589,89	23	13567
Науковий співробітник	10005	480,62	23	11054
Тестувальник	8988	717,85	21	15074
Лаборант	7040	685,2	22	15074
Всього				54769

Основна заробітна плата робітників: витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт розраховують за формулою:

$$Z_p = \sum_{i=1}^n C_i * t_i,$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M * K_i * K_C}{T_p * t_{зм}},$$

де M_M – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати, грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б);

K_C – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати. (табл. Б.1, додаток Б)

T_p – середня кількість робочих днів в місяці, приблизно $T_p = 21 \dots 23$ дні;

$t_{зм}$ – тривалість зміни, год.

Таблиця 4.5 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника, грн
Координація роботи партнерів	7	3	1.5	84,27	589,89

Продовження таблиці 4.5

Участь у впровадженні результатів проведених досліджень та розробок	7	2	1,5	68,66	480,62
Тестування	7	4	1,5	102,55	717,85
Проведення дослідів	6	4	1,5	114,20	685,2
Всього					2473,56

$$C_1 = \frac{6700 * 1,35 * 1,5}{23 * 7} = \frac{13567,5}{161} = 84,27;$$

$$C_2 = \frac{6700 * 1,1 * 1,5}{23 * 7} = \frac{11055}{161} = 68,66;$$

$$C_3 = \frac{6700 * 1,5 * 1,5}{21 * 7} = \frac{15075}{147} = 102,55;$$

$$C_4 = \frac{6700 * 1,5 * 1,5}{22 * 6} = \frac{15075}{132} = 114,20;$$

$$\begin{aligned} Z_p &= \sum_{i=1}^n C_i * t_i = 84,27 * 7 + 68,66 * 7 + 102,55 * 7 + 114,20 * 6 \\ &= 589,89 + 480,62 + 717,85 + 685,2 = 2473,56; \end{aligned}$$

Додаткова заробітна плата: розраховується як 10-12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_{\text{дод}} = (Z_o + Z_p) * \frac{H_{\text{дод}}}{100\%} = (53769 + 2473) * \frac{10\%}{100\%};$$

$$З_{\text{дод}} = 5724,2;$$

4.2.2. Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховується як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_{\text{н}} = (З_{\text{о}} + З_{\text{р}} + З_{\text{дод}}) * \frac{Н_{\text{зп}}}{100\%},$$

Де $H_{\text{зп}}$ – норма нарахування на заробітну плату.

$$З_{\text{н}} = (54769 + 2473 + 5724) * \frac{22\%}{100\%},$$

$$З_{\text{н}} = 13852,2$$

4.2.3 Сировина та матеріали

При виконанні науково-дослідницької роботи необхідності сировині та матеріалах не було.

4.2.4 Розрахунок витрат на комплектуючі

При виконанні науково-дослідницької роботи всі роботи проводились з використанням лише наявних комплектуючих.

4.2.5 Спецустаткування для наукових робіт

При виконанні науково-дослідницької роботи всі роботи проводились з використанням лише наявного спеціального обладнання.

4.2.6 Програмне забезпечення для наукових робіт

До балансової вартості програмного забезпечення входять витрати на його інсталяцію, тому ці витрати беруться додатково в розмірі 10-12% від вартості програмного забезпечення.

Балансову вартість програмного забезпечення розраховують за формулою:

$$B_{\text{прг}} = \sum_{i=1}^k C_{\text{іпрг}} * C_{\text{прг.і}} * K_i,$$

де $C_{\text{іпрг}}$ – ціна придбання одиниці програмного засобу цього виду, грн;

$C_{\text{прг.і}}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань програмних засобів.

Отримані результати зведено до таблиці 4.6.

Таблиця 4.6 – Витрати на придбання програмних засобів кожного виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Sublime Text	1	1000	1000
Host Website	1	1200	1200

Продовження таблиці 4.6

Domain Website	1	480	480
DataBase	1	2000	2000
Всього			5148

$$B_{\text{прг}} = (1000 * 1 * 1,1) + (1200 * 1 * 1,1) + (480 * 1 * 1,1) + (2000 * 1 * 1,1) \\ = 1100 + 1320 + 528 + 2200 = 5148;$$

4.2.7 Амортизація обладнання, програмних засобів та приміщень

При виконанні науково-дослідницької роботи всі роботи проводились з використанням лише наявного обладнання.

4.2.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію розраховуються за формулою:

$$B_e = \sum_{i=1}^k \frac{W_{yi} * t_i * C_e * K_{\text{впі}}}{\eta_i},$$

де W_{yi} – встановлена потужність обладнання на певному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії);

$K_{\text{впі}}$ – коефіцієнт, що враховує використання потужності;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

Проведені розрахунки зведено до таблиці 4.7.

Таблиця 4.7 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Системний блок	0.1	184	142,6
Монітор	0.02	184	28,5
Всього			171

$$B_e = \frac{0,1 * 184 * 6,2 * 100\%}{80\%} + \frac{0,02 * 184 * 6,2 * 100\%}{80\%} = 142,6 + 28,5 = 171;$$

4.2.9 Службові відрядження

При виконанні науково-дослідницької роботи за даною темою, згідно плану, ніякі службові відрядження не проводились.

4.2.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

У виконанні науково-дослідницької роботи сторонні організації участі не приймали.

4.2.11 Інші витрати

Витрати за статтею «Інші витрати» розраховуються як 50-100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_B = (3_o + 3_p) * \frac{H_{iB}}{100\%},$$

де H_{iB} – норма нарахування за статтею «Інші витрати».

$$I_B = (54769 + 2473) * \frac{75\%}{100\%} = 42931;$$

4.2.12 Накладні витрати

Витрати за статтею «Накладні витрати» розраховується як 100-150% від суми основної заробітної плати дослідників та робітників за формулою:

$$V_{\text{нзв}} = (Z_o + Z_p) * \frac{H_{\text{нзв}}}{100\%},$$

Де $H_{\text{нзв}}$ – норма нарахування за статтею «Накладні (загальновиробничі) ви-трати».

$$V_{\text{нзв}} = (54769 + 2473) * \frac{125\%}{100\%} = 71552;$$

Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$V_{\text{заг}} = Z_o + Z_p + Z_{\text{дод}} + Z_{\text{н}} + V_{\text{прг}} + V_e + I_v + V_{\text{нзв}},$$

$$V_{\text{заг}} = 54769 + 2473 + 5724,2 + 13852 + 5148 + 171 + 42931 + 71552,$$

$$V_{\text{заг}} = 196620;$$

Загальні витрати ЗВ на завершення науково-технічної роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{V_{\text{заг}}}{\eta},$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-

дослідної роботи. Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то $\eta = 0,1$; технічного проектування, то $\eta = 0,2$; розробки конструкторської документації, то $\eta = 0,3$; розробки технологій, то $\eta = 0,4$; розробки дослідного зразка, то $\eta = 0,5$; розробки промислового зразка, то $\eta = 0,7$; впровадження, то $\eta = 0,9$.

$$ЗВ = \frac{В_{\text{заг}}}{\eta} = \frac{196620}{0,4} = 491550;$$

4.3 Оцінювання економічної ефективності науково-технічної розробки

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора чистого прибутку.

Результати дослідження проведені за темою «Інформаційна система онлайн помічника для продажу товарів» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

ΔN – Збільшення кількості споживачів яким надається відповідна інформаційна послуга у період часу, що аналізується;

Таблиця 4.7 – Збільшення споживачів програмного засобу

Показник	1й рік	2й рік	3й рік	4й рік
Збільшення кількості споживачів, осіб	5000	10000	8000	7000

N – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження нової науково-дослідної розробки, прийємо 25000 осіб;

C_0 - вартість послуги у році до впровадження інформаційної системи, прийmemo 250,00 грн;

$\pm\Delta C_0$ – зміна вартості послуги від впровадження результатів, прийmemo 50,00 грн;

Можливе збільшення чистого прибутку у потенційного інвестора ΔP_i для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-дослідної розробки, розраховуємо за формулою:

$$\Delta P_i = (\pm\Delta C_i * N + C_0 * \Delta N) * \lambda * \rho * \left(1 - \frac{\vartheta}{100}\right),$$

λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2022 році ставка податку на додану вартість становить 20%, а коефіцієнт $\lambda = 0,8333$;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту. Приймемо $\rho = 0,35$;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2022 році $\vartheta = 18\%$.

Збільшення чистого прибутку 1-го року:

$$\Delta P_1 = (50,00 * 25000,00 + 300 * 5000) * 0,83 * 0,35 * \left(1 - \frac{18\%}{100}\right);$$

$$\Delta P_1 = 655077,00 \text{ грн};$$

Збільшення чистого прибутку 2-го року:

$$\Delta P_2 = (50,00 * 25000,00 + 300 * 15000) * 0,83 * 0,35 * \left(1 - \frac{18\%}{100}\right);$$

$$\Delta P_2 = 1369707,00 \text{ грн};$$

Збільшення чистого прибутку 3-го року:

$$\Delta P_3 = (50,00 * 25000,00 + 300 * 23000) * 0,83 * 0,35 * \left(1 - \frac{18\%}{100}\right);$$

$$\Delta P_3 = 1941411,00 \text{ грн};$$

Збільшення чистого прибутку 4-го року:

$$\Delta\Pi_4 = (50,00 * 25000,00 + 300 * 30000) * 0,83 * 0,35 * \left(1 - \frac{18\%}{100}\right);$$

$$\Delta\Pi_4 = 2118354,00 \text{ грн};$$

Приведена вартість збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-дослідної розробки:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t}$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,1$;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році;

$$ПП = \frac{655077,00}{1,1^1} + \frac{1369707,00}{1,1^2} + \frac{1941411,00}{1,1^3} + \frac{2118354,00}{1,1^4} =;$$

$$ПП = 595524,54 + 1131989,26 + 1459707,52 + 1450927,4;$$

$$ПП = 4638148,7 \text{ грн};$$

Величина початкових інвестицій , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{\text{інв}} * ЗВ,$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науко-во-технічної розробки та її комерціалізацію. Це можуть бути витрати на під-готовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{\text{інв}} = 2$;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 491550,00 грн.

$$PV = 2 * 491550,00 = 983100,00 \text{ грн};$$

Абсолютний економічний ефект або чистий приведений дохід для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{abc} = \text{ПП} - PV,$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, приймаємо 4638148,7 грн;

PV – теперішня вартість початкових інвестицій, 983100,00 грн.

$$E_{abc} = 4638148,7 - 983100,00 = 3655048,00 \text{ грн};$$

Внутрішня економічна дохідність інвестицій , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-

технічної розробки:

$$E_B = \sqrt[T_{ж}]{1 + \frac{E_{abc}}{PV}} - 1,$$

де E_{abc} – абсолютний економічний ефект вкладених інвестицій, 3655048,00 грн;

PV – теперішня вартість початкових інвестицій, 983100,00 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 4 роки.

$$E_B = \sqrt[4]{1 + \frac{3655048,00}{983100,00}} - 1 = \sqrt[4]{4,71} - 1 = 1,47 - 1 = 0,47;$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій визначається за формулою:

$$\tau_{\text{мін}} = d + f,$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2022 році в Україні $d = 0,1$;

f – показник, що характеризує ризикованість вкладення інвестицій; зазвичай величина $f = 0,1$.

$$\tau_{\text{мін}} = 0,1 + 0,1 = 0,2;$$

$0,2 < 0,33$ свідчить про те, що внутрішня економічна дохідність інвестицій, які можуть бути вкладені потенційним інвестором у впровадження

та комерціалізацію науково-дослідної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Інформаційна система онлайн помічника для продаже товарів» доцільно.

Період окупності інвестицій, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_B},$$

де E_B – внутрішня економічна дохідність вкладених інвестицій

$$T_{ок} = \frac{1}{0,47} = 2,12;$$

$T_{ок} < 3$ – х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

4.4 Висновок до розділу 4

У цьому розділі було проведено науковий аудит науково-дослідної роботи. Було здійснено оцінювання наукового ефекту. Визначено новизну роботи, рівень її теоретичного опрацювання, перспективність, рівень розповсюдження результатів, можливість реалізації.

Далі було проведено ряд розрахунків, пов'язаних з науково-дослідною роботою, серед яких: витрати на оплату праці, на соціальні заходи, на сировину та матеріали, також витрати на комплектуючі та інші витрати.

Результатом цього розділу є повний розрахунок вартості реалізації даної науково-дослідної роботи.

ВИСНОВКИ

Відповідно до аналізу предметної області, а саме дослідження ринку інформаційних технологій онлайн помічника для продажу товарів, було прийнято рішення розробки онлайн помічника у вигляді системи рекомендацій, яка буде спрощувати продаж товарів для продавців, та полегшувати пошук та покупку для покупців.

Аналіз аналогів дав зрозуміти, яка саме система необхідна, та що потрібно для її удосконалення. Було вирішено питання актуальності та працездатності онлайн помічника. Запропоновано використання нечіткого методу рекомендацій товарів, за рахунок використання бібліотек NumPy та Pandas, які є необхідними для аналізу масиву даних, що дозволять створити систему персоналізованих рекомендацій.

Було удосконалено інформаційну технологію онлайн помічника для продажу товарів, основуючись на результатах аналізу аналогів. Удосконалено рекомендацію товарів, за рахунок підвищення якості, та швидкості опрацювання. Це стало можливим завдяки функціональним можливостям системи рекомендацій.

Функціональні можливості системи рекомендацій:

- Аналіз масивів даних;
- Розподіл даних за категоріями;
- Пріоритизація завдань;
- Зручний інтерфейс;
- Користувацькі налаштування;
- Збільшення точності рекомендацій;
- Збільшення якості рекомендацій;
- Збільшення кількості користувачів;
- Резервне копіювання;
- Штучне навчання.

Було розроблено структуру інформаційної технології онлайн помічника для продажу товарів, та спроектовано схему алгоритму роботи розроблюваного програмного засобу. Розроблено алгоритм функціонування інформаційної технології

Здійснено проектування та розробку інформаційної технології онлайн помічника для продажу товарів. Обґрунтовано вибір мови програмування та середовища. Програмна реалізація онлайн помічника для продажу товарів у середовищі Visual Studio, мовою програмування Python, з використанням бібліотек NumPy та Pandas для роботи з масивами даних користувачів.

Проведено тестування роботи інформаційної технології онлайн помічника для продажу товарів та проаналізовано її результати. Тестування програми підтвердило очікувані результати та правильність роботи, довівши підвищення точності прогнозування.

Здійснено економічне обґрунтування доцільності розробки інформаційної технології онлайн помічника для продажу товарів. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, знайдено термін окупності витрат для виробника та економічний ефект для споживача при використанні такої розробки. В результаті аналізу розрахунків можна зробити висновок, що розробка у виробництві та використанні дешевша за аналог і є висококонкурентноспроможною. Період окупності складе 2,12 років. Загальні витрати становлять 491550,00 грн.

В результаті даної роботи отримано програмний продукт – систему рекомендацій. Мета роботи якого, спрощення процедури рекомендації та покупки товарів шляхом створення нечіткої системи підтримки прийняття рішень при виборі товарів. Отже, мета магістерської кваліфікаційної роботи досягнута.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бортновський О. С., Озеранський В. С. «Інформаційна система онлайн помічника для продажу товарів», в Матеріали конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2022)», Вінниця, 2022, [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2023/paper/view/16864>
2. Recommendation system in E-Commerce, [Електронний ресурс]. Режим доступу: – <https://www.be-terna.com/insights/recommendation-systems-in-e-commerce-whats-the-thing-youve-never-known-but-always-wanted-to>
3. What are recommendation system? , [Електронний ресурс]. Режим доступу: – <https://www.ionos.com/digitalguide/online-marketing/online-sales/how-to-use-recommendation-systems-in-e-commerce/>
4. Все що потрібно знати про чат-ботів, [Електронний ресурс]. Режим доступу: – <https://toplead.com.ua/ru/blog/id/fakty-o-chat-botah-168/>
5. Recommender System in E-Commerce. Abstract and figures, [Електронний ресурс]. Режим доступу: – https://www.researchgate.net/publication/2507550_Recommender_Systems_in_E-Commerce
6. E-Commerce recommendation system engine with collaborative filtering, [Електронний ресурс]. Режим доступу: – <https://medium.com/web-mining-is688-spring-2021/e-commerce-recommendation-engine-with-collaborative-filtering-cb19cd542c18>
7. Статистика Google Trends, [Електронний ресурс]. Режим доступу: – <https://trends.google.com/trends/explore?q=chatbot&geo=US>
8. A Content-Based Recommender for E-Commerce, [Електронний ресурс]. Режим доступу: – <https://towardsdatascience.com/a-content-based-recommender-for-e-commerce-web-store-7554b5b73eac>
9. Що таке онлайн асистент. Визначення та керування, [Електронний ресурс]. Режим доступу: –

- <https://sendpulse.ua/ru/support/glossary/chatbot>
10. Product recommendation engine for E-Commerce, [Електронний ресурс].
Режим доступу: – <https://datrics.ai/product-recommendations-engine-for-e-commerce>
 11. Про популярність чат-ботів, [Електронний ресурс]. Режим доступу: –
<https://businesspartnermagazine.com/why-chatbots-becoming-popular-onlinemarketing/#:~:text=Realtime%20engagement%20and%20interaction,increases%20brand%20loyalty%20an>
 12. Recommendation system for E-Commerce Systems an overview, [Електронний ресурс]. Режим доступу: –
<https://iopscience.iop.org/article/10.1088/1742-6596/1897/1/012024/pdf>
 13. Чат-боти: Для чого створювати, кому використовувати, [Електронний ресурс]. Режим доступу: – <https://blog.click.ru/growthhacking/chat-bot-na-sajte-zachem-sozdavat-komu-ispolzovat-i-kak-nastroit/>
 14. What is an e-commerce recommendation system? Everything on the topic! , [Електронний ресурс]. Режим доступу: –
<https://www.smarthint.co/en/What-is-a-recommendation-system%3F/>
 15. Онлайн помічник для бізнесу, [Електронний ресурс]. Режим доступу: –
<https://ivbots.store>
 16. Інтернет-магазин «Rozetka» , [Електронний ресурс]. Режим доступу: –
<https://www.muvi.com/blogs/how-to-integrate-recommendation-engine-with-e-commerce-platforms.html>
 17. Платіжна система LiqPay, [Електронний ресурс]. Режим доступу: –
<https://www.liqpay.ua/uk/>
 18. Хостинг сервіс, [Електронний ресурс]. Режим доступу: –
<https://vercel.com/dashboard>
 19. How to integrate recommendation engine with e-commerce platforms, [Електронний ресурс]. Режим доступу: –
<https://www.muvi.com/blogs/how-to-integrate-recommendation-engine-with-e-commerce-platforms.html>

ДОДАТКИ

Додаток А (обов'язковий)

Результат перевірки на плагіат в онлайн-системі UNICHECK



Имя пользователя:
Озеранський В.С. КН

ID проверки:
1013270333

Дата проверки:
11.12.2022 19:59:59 EET

Тип проверки:
Doc vs Internet + Library

Дата отчета:
11.12.2022 20:01:40 EET

ID пользователя:
62038

Название файла: 122МКР-БортновськийОС2022

Количество страниц: 42 Количество слов: 6188 Количество символов: 46412 Размер файла: 715.85 KB ID файла: 1013029113

17.7% Совпадения

Наибольшее совпадение: 17.7% с Интернет-источником (<https://essuir.sumdu.edu.ua/bitstream/123456789/78592/1/Don>)

17.7% Источники из Интернета 1 Страница 44

Не найдены источники из Библиотеки

0% Цитат

Исключение цитат выключено

Исключение списка библиографических ссылок выключено

3.81% Исключений

Некоторые источники исключены автоматически (фильтры исключения: количество найденных слов меньш...

0.94% Исключений из Интернета 17 Страница 45

3.68% Исключенного текста из Библиотеки 128 Страница 45

Додаток Б (обов'язковий)**Лістинг програми**

```
import load_data
import pandas as pd
import numpy as np
from fuzzy_sets import Age, GIM
import gim
import operator

a = Age()
g = GIM()

def euclidian(A, B):
    return np.linalg.norm(np.array(A) - np.array(B))

def dist_func(a, b, A, B):
    return abs(a - b) * euclidian(A, B)

def fuzzy_distance(ui, uj):
    fuzzy_v = [0] * 21
    for i in range(0, 19):
        ui_gim = [g.very_bad(ui[i]), g.bad(ui[i]), g.average(ui[i]), g.good(ui[i]), g.very_good(ui[i]),
                  g.excellent(ui[i])]
        uj_gim = [g.very_bad(uj[i]), g.bad(uj[i]), g.average(uj[i]), g.good(uj[i]), g.very_good(uj[i]),
                  g.excellent(uj[i])]
        fuzzy_v[i] = dist_func(ui[i], uj[i], ui_gim, uj_gim)
    i = 19
    ui_gim = [a.young(ui[i]), a.middle(ui[i]), a.old(ui[i])]
```

```

uj_gim = [a.young(uj[i]), a.middle(uj[i]), a.old(uj[i])]
fuzzy_v[i] = dist_func(ui[i], uj[i], ui_gim, uj_gim)
# додаємо user_id другого користувача
fuzzy_v[20] = uj['user_id']
# виводимо fuzzy_v
return fuzzy_v

```

```

def recommend(neighbours, testing):
    ms = 0
    for key, item in testing.iterrows():
        ar,m_id = item['rating'],item['book_id']
        n_ratings = []
        for i in neighbours:
            temp = df.loc[df['user_id'] == i].loc[df['book_id'] == m_id]
            for k, it in temp.iterrows():
                n_ratings.append(it['rating'])
        pr = float(sum(n_ratings)) / len(n_ratings) if len(n_ratings) else 0
        ms += abs(pr-ar)
    print (testing.shape[0])
    return ms/testing.shape[0]

```

```

# виводимо fuzzy_dist(35, 40, np.array([3, 4, 5]), np.array([4, 5, 6]))
# Деталі даних
# users_cols = 'user_id', 'age', 'sex', 'occupation', 'zip_code'
# ratings_cols = 'user_id', 'book_id', 'rating', 'unix_timestamp'
# i_cols = ['book_id', 'book_title', 'release date', 'video release date', 'IMDb URL', 'unknown',
'Action', 'Adventure',
# 'Animation', 'Children\s', 'Comedy', 'Crime', 'Documentary', 'Drama', 'Fantasy',
# 'Film-Noir', 'Horror', 'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Thriller', 'War', 'Western']
# All in one DataFrame

```

```

mr_ur = pd.merge(load_data.users, load_data.ratings, on='user_id')
df = pd.merge(mr_ur, load_data.items, on='book_id')
m_cols = ['unknown', 'Action', 'Adventure',
          'Animation', 'Children\s', 'Comedy', 'Crime', 'Documentary', 'Drama', 'Fantasy',
          'Film-Noir', 'Horror', 'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Thriller', 'War', 'Western',
          'age',
          'user_id']
model_data_au = pd.DataFrame(columns=m_cols)
feature_row = pd.DataFrame(columns=m_cols)
model_data_pu = pd.DataFrame(columns=m_cols)
# Користувачі, які оцінили книги щонайменше 60 книгам
top_users = load_data.df.groupby('user_id').size().sort_values(ascending=False)[:497]

for i in range(0, 5):
    # active_users and passive_users - pd.Series()
    active_users = top_users.sample(frac=0.10)

    # training_active_users = active_users.sample(frac=0.34)
    # testing_active_users = active_users.drop(training_active_users.index)

    passive_users = top_users.drop(active_users.index)
    # To change this # active users
    tau_data = df.loc[df['user_id'].isin(active_users)][:10]
    index = np.arange(0, tau_data.shape[0])
    i = 0
    for key, value in tau_data.iterrows():
        user_ui_books = df.loc[df['user_id'] == value['user_id']]

        training_user_books = user_ui_books.sample(frac=0.34)
        feature_array = gim.gim_final(training_user_books, value['user_id'])

    # print 'GIM array', feature_array

```

```

feature_array[19], feature_array[20] = value['age'], value['user_id']
# print feature_array.shape
model_data_au.loc[i] = feature_array
i = i + 1
# print model_data_au
# print model_data_au
# Working with passive users
# change # of passive users

i = 0
pu_data = df.loc[df['user_id'].isin(passive_users)][:10]

for key, value in pu_data.iterrows():
    user_ui_books = df.loc[df['user_id'] == value['user_id']]
    feature_array_p = gim.gim_final(user_ui_books, value['user_id'])
    # print 'GIM array', feature_array
    feature_array_p[19], feature_array_p[20] = value['age'], value['user_id']
    # print feature_array.shape
    model_data_pu.loc[i] = feature_array_p
    i = i + 1
# print model_data_au
# print model_data_pu

# fuzzy_df = pd.DataFrame(columns=m_cols)
fuzzy_vec = []
error = []
for key, value in model_data_au.iterrows():
    i = 0
    for key1, value1 in model_data_pu.iterrows():
        fuzzy_vec.append(fuzzy_distance(value, value1))
        # print value[i], value1[i]
        fuzzy_vec[i] = [(sum(x for x in fuzzy_vec[i][:-1]))**0.5, fuzzy_vec[i][-1]]

```

```
i = i + 1
# print fuzzy_vec
neighbours = [n[1] for n in sorted(fuzzy_vec, key=operator.itemgetter(0),
reverse=True)][:30] # taking top 30
# print neighbours
testing_user = df.loc[df['user_id'] == value['user_id']].sample(frac=0.66)
e = recommend(neighbours, testing_user)
print (e)
error.append(e)
MAE = sum(error) / len(error)
print (MAE)
```

Додаток В (обов'язковий)

80

Додаток В (обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ОНЛАЙН ПОМІЧНИКА ДЛЯ
ПРОДАЖУ ТОВАРІВ

Виконав: студент 2-го курсу,
групи 2КН-21м

спеціальності 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки, спеціальності)

О Бортновський О.С. _____
(прізвище та ініціали)

Керівник: к.т.н., ст. викладач каф. КН
В Озеранський В. С. _____
(прізвище та ініціали)

« 15 » 12 2022 р.

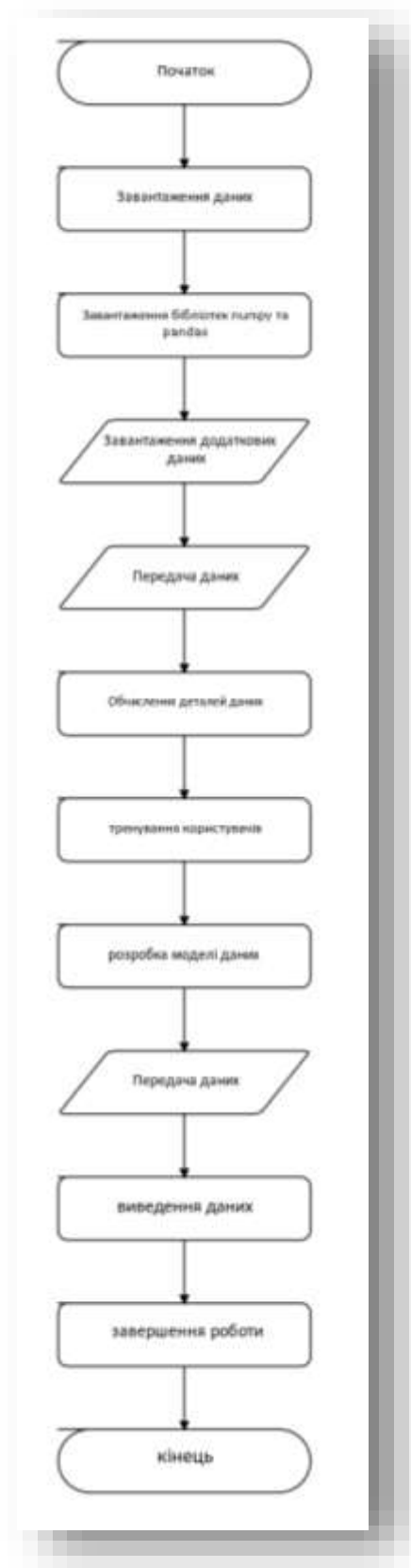


Рисунок В.1 – Загальний алгоритм функціонування системи рекомендації товарів

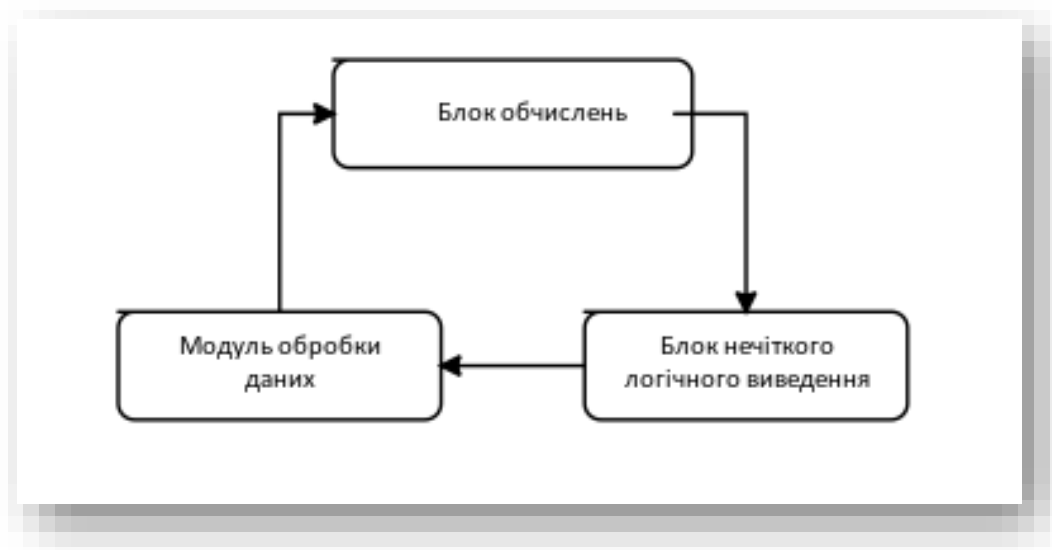


Рисунок В.2 – Загальна структура системи рекомендації товарів

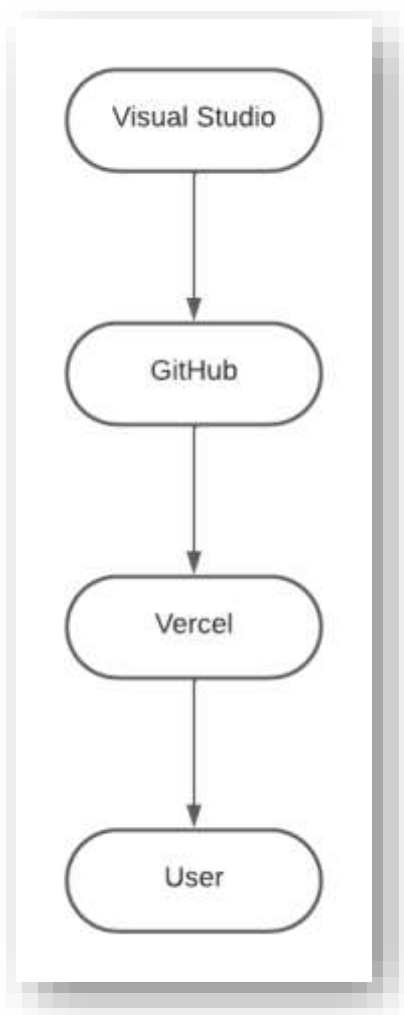


Рисунок В.3 – Загальна схема роботи системи рекомендації товарів

Додаток Г (довідковий)

Інструкція користувача

1. Завантажити отриманий архів “fgrs.zip”
2. Якщо у вашому інтернет-магазині назви не стандартні, та не збігаються, необхідно їх замінити.
3. Готову систему необхідно імпортувати на хостинг вашого проекту. Зробити це можливо через FTP-панель.
4. Наступним пунктом буде підключення. Для цього необхідно використати фреймворк. Наприклад «PyPython». Для його використання необхідно вставити наступний код у тіло файлу html : `<link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />`
`<script defer src="https://pyscript.net/latest/pyscript.js"></script>`
5. Далі необхідно перепустити роботу хостингу, щоб скрипт заробив.